A COMPUTER BASED DATA MANAGEMENT SYSTEM FOR AIR FORCE WAR RESERVE MATERIEL (WRM) VEHICLE MANAGEMENT

THESIS

Robert S. Thomas
First Lieutenant, USAF

AFIT/GLM/LSM/88S-70

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

(2)

AFIT/GLM/LSM/88S-70

A COMPUTER BASED DATA MANAGEMENT SYSTEM FOR AIR FORCE WAR RESERVE MATERIEL (WRM) VEHICLE MANAGEMENT

THESIS

Robert S. Thomas
First Lieutenant, USAF

AFIT/GLM/LSM/88S-70

Approved for public release; distribution unlimited

The contents of the document are technically accurate, and no sensitive items, detrimental ideas, or deleterious information is contained therein. Furthermore, the views expressed in the document are those of the author and do not necessarily reflect the views of the School of Systems and Logistics, the Air University, the United States Air Force, or the Department of Defense.

AFIT/GLM/LSM/88S-70

A COMPUTER BASED DATA MANAGEMENT SYSTEM FOR AIR FORCE WAR RESERVE MATERIEL (WRM) VEHICLE MANAGEMENT

THESIS

Presented to the Faculty of the School of Logistics
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree
Master of Science in Logistics Management

Robert S. Thomas, B.S.
First Lieutenant, USAF

September 1988

Preface

The purpose of this research was to develop a software database management system to manage Air Force WRM vehicles. Two software programs were instrumental in completion of the WRM vehicle management program, Ashton-Tate™ dBase III PLUS™ database management software, and the dBfast™ compiler. I wish to thank the dBfast technical advisors for their many hours of assistance in applying their compiler to my application.

I wish to present a special note of thanks to Captain James R. VanScotter for providing me with the idea for this thesis, and a tremendous amount of encouragement and technical assistance. No project of this magnitude can reach completion without someone who knows when to lead or encourage. I would like to thank my thesis advisor Lieutenant Colonel John M. Halliday for giving me the push when I needed it, and good advice throughout. Finally, I would like to dedicate this thesis to my wife Pat, and my three children, without whom this effort would have been meaningless and certainly more tiresome.

Robert S. Thomas

Table of Contents

List of Figures

List of Programs

Abstract

The purpose of this research was to determine the feasibility of developing a microcomputer based system for use by transportation personnel to manage the War Readiness Materiel (WRM) fleet. This research determined user requirements, developed a prototype system, and validated the prototype system through pre-field testing.

Coordination with Air Force Logistics Management Center (AFLMC) transportation personnel, and HQ PACAF/LGT personnel was established to develop a field testing program for successful implementation of the WRM Vehicle Management System. The prototype system permits manual and automated input of data. The system is designed to accept automated input from the Vehicle Integrated Management System (VIMS) and the AFLMC Vehicle Automated Management System (VAMS).

The WRM Vehicle Mangagement System provides capability for vehicle dispersal/distribution management, scheduled actions management, release case management, and a variety of reports for the whole fleet or a subset of the fleet. The end product is a computer program on a single 5 1/4 inch "floppy disk" that will operate on IBM™ or Zenith™ micro-computers. The program was compiled to provide stand alone capability to limit the cost of implementation.

A COMPUTER BASED DATA MANAGEMENT SYSTEM FOR AIR FORCE WAR RESERVE MATERIEL (WRM) VEHICLE MANAGEMENT

I. Introduction

General Background

"The conduct of war is the art and science of using military force with other instruments of national power to achieve victory (1:1)." The need to rapidly project power, in support of national objectives, to potentially any geographic area of the world has increased logistic requirements beyond those of past wars.

> "The next war may not resemble past wars. Technological advances can overpower tradition and create new and unanticipated environments for combat operations (2:3.5)."

Vietnam, the United States most recent war fought on a large scale, underscored the logistic burden that distance and varying combat conditions place upon the Defense Transportation System.

> According to Gen William C. Westmoreland, commander of Military Assistance Command, Vietnam (MACV) from 1964 to 1968, it took him three years to obtain the personnel, to establish his transportation system, and secure the bases he needed to go on the offensive against the Viet Cong in 1967. General Westmoreland spent these three years building a logistics base to support the kind of operations he wanted to carry out - airmobile assaults dependent upon complex machinery (3:46).

The Defense Transportation System ultimately delivered 17,200,000 tons of cargo in support of the war effort (4:85). The Vietnam War lasted several years with gradual stages of

buildup allowing the deficiencies in the transportation system to be overcome. Current military doctrine recognizes that a much shorter time period for the buildup of hostilities will be likely for any wars fought in the European or Pacific theaters (5:8). America's current defense is based on a policy of deterrence implemented by a forward defense strategy (15:26-27). This forward defense strategy translates into the need for strategic mobility. "Strategic mobility is a triad of airlift, sealift and prepositioning (16:20)." The Military Sealift Command is the single manager of sealift for the Department of Defense.

> More than 90 percent of equipment and supplies needed to sustain a war effort must be carried by ship....As recently as last year, the Department of Defense determined that, under certain conditions, our ability to meet the initial deployment requirements of war would be "marginally inadequate" in the future if programs existing in 1983, remain unaltered (5:15).

The initial surge requirements to rapidly deploy and replenish combat forces must be met by airlift. The Military Airlift Command (MAC) is the single manager for all Department of Defense airlift requirements. MAC has indicated that multiple wartime scenarios could generate a requirement for up to 90 million ton miles per day (MTM/D). A recent Government Accounting Office study identified that the total capacity of civil and military aircraft would leave a shortfall of 31.2 MTM/D of cargo capacity (6:8). This shortfall exists despite the fact that the MAC projected cargo requirements have been reduced by the existence of prepositioned equipment.

The inability of sealift and airlift to meet all of the surge strategic mobility requirements has increased the reliance upon prepositioning. "The Army, recognizing the need for immediately available equipment for deploying forces, began a prepositioning process in the early 70's...called POMCUS (Prepositioning of Materiel configured to Unit Sets) (3:97)." Air Force war planners also began to preposition equipment, weapons, and ammunition. The remainder of this thesis will focus upon the War Reserve Materiel (WRM) vehicle program administered by the Air Force.

The number of WRM vehicles prepositioned in Europe and the Far East has grown from 1530 in 1977 to the current level of 12,000 ultimately peaking at 26,000 when the full authorized strength is achieved (6:891). "The Air Force possesses more than 148,000 vehicles to perform various missions around the world (9:1)." Thus, Air Force bases maintain both a peactime and wartime vehicle fleet. The responsibility for maintaining and managing both vehicle fleets is delegated to base transportation squadrons (14:9). The typical base transportation squadron is divided into four branches: Traffic Management, Plans and Programs, Vehicle Maintenance, and Vehicle Operations. The Vehicle Maintenance branch has the responsibility for maintaining vehicles in a serviceable condition and Vehicle Operations has responsibility for management of the WRM and peacetime fleets (14:9). The Fleet Management section of Vehicle Operations has the responsibility to ensure that WRM and peacetime

requirements are met (13:284). The goal of the Air Force WRM vehicle program is to preposition all vehicles needed to support the additive wartime requirements of committed forces, except those vehicles unique to a unit's mission and identified as mobility assets to move with the deploying unit (11:38). Air Force vehicle authorizations are classified by a "use code" that describes the vehicle's intended use and its justification for authorization. Vehicle use codes range from A to U. WRM vehicles are classified with a use code D and divided into two groups, pure and integrated WRM. Pure WRM vehicles are by definition pre-positioned to support forces scheduled to be in place within 60 days of the initiation of a war plan (13:270-271). Integrated WRM vehicles, are vehicles that cannot be placed into storage because of their unique operating design, and are therefore integrated into the unit's peacetime fleet to maintain their integrity and serviceability (13:271). Examples of integrated WRM vehicles include Materiel Handling Equipment like fork-lifts and other vehicles with a unique operating design such as firetrucks. WRM vehicles are subjected to a continual process of rotation into storage, removal from storage for scheduled exercising and maintenance action, and rotation with peacetime assets to ensure optimum aging of the entire base fleet (13:279). WRM vehicle fleets range in size from a few to 1200 vehicles (10:15). The number of personnel assigned to manage the WRM fleet at any base, varies from

none where the fleet is small, to five at bases with large fleets. The rigidly enforced requirement to maintain the separate integrity of each fleet, without commingling of assets (other than integrated WRM vehicles) has forced the Vehicle Management to maintain two completely separate reporting and scheduling systems. Fleet Management personnel have used a mixture of manual scheduling techniques, computer generated reports, and some personal computer database programs to manage the WRM fleet. A recently developed Air Force Logistics Management Center (AFLMC) personal computer database program has eased the burden of managing the vehicle fleets. The Vehicle Asset Management System (VAMS) is designed to help fleet managers keep track of vehicle authorizations and assignments for both the Peacetime and WRM fleets. The VAMS system does not however eliminate the management burden associated with maintaining, exercising, scheduling, and reporting activities for the WRM vehicle fleet.

Specific Problem

The recent explosive growth in the size of the WRM fleet coupled with an equally increased reliance upon its wartime capability, has placed an overwhelming burden upon the potpourri of base vehicle management systems in use. Beginning in 1984, Air Force Inspector General Functional Management Inspections identified numerous problems with the management procedures for both the WRM and Peacetime fleets.

The 1984 Air Force "Functional Management Inspection of Transportation Contingency Planning" described the following WRM vehicle management problems pertinent to this research:

1. Transportation reception plans were often poorly staffed, vague, and ill-defined because insufficient time was available to additional duty planners... Adequate time could not be devoted to the myriad of tasks necessary to complete effective contingency planning...The long term impact of such situations could be a failure of transportation to support the mission (8:7).

2. Few base-level transportation planners were aware of transportation personnel or vehicles which might be deployed into their bases as part of nontransportation unit type codes (UTC)...A few major commands had attempted to determine the total scope of personnel and vehicles in any deployed UTC by suggesting a computer sort according to specialty codes and vehicle stock numbers but had not achieved any notable success (8:8-9).

3. Another weakness observed within transportation units was the lack of a concept of transportation operations for the period immediately following force reception and Collocated base (COB) activation ...Specifically, there had been little contemplation of the sourcing of personnel and vehicles which might be required to deliver or reposition sometimes huge numbers of personnel and tons of deployed equipment (8:10).

4. War Reserve Materiel (WRM) vehicle management required improvement to resolve deficiencies in storage, movement plans, manpower, and technical data support (8:15).

In 1986, an Air Force IG "Functional Management Inspection of USAF Vehicle Authorization, Acquisition, and Allocations Programs" continued the audit trail of WRM vehicle management problems by identifying problems more specific to actual fleet management activities. A summary of the pertinent problems are as follows:

1. Lack of functional consolidation between the Base-Level Registered Equipment Management System (REMS) and Fleet Management resulted in duplication of effort and multiple source documents containing erroneous information...Formulation of these source documents were accomplished through a time-consuming manual editing process, greatly slowing accurate vehicle accountability...lack of a summarized change listing on vehicle source documents increased the time required to reconcile new document updates (9:9).

2. Lack of Air Force standardized vehicle management micro-computer programs and training resulted in manually produced obsolete data and analysis (9:11).

In June of 1986, Pacific Air Force (PACAF) headquarters requested support from the Air Force Logistics Management Center (AFLMC) to develop a functional description for a WRM vehicle management system. AFLMC representatives responded by scheduling visits to United States Armed Forces Europe (USAFE) bases (HQ-USAFE-Germany, Ramstein AB-Germany, Sembach AB-Germany, Mildenhall AB-England) during 21 September 1986 to 3 October 1986 and PACAF bases (HQ PACAF-Hawaii, Osan Air Base(AB)-Korea, Kunsan AB-Korea, Suwon AB-Korea) during 25 October 1986 to 4 November 1986. The primary purpose of these visits, was to meet with transportation personnel, and validate what type of a vehicle management system would resolve any problems that might exist. Both visits confirmed that many of the problems identified in previous IG inspections continued to exist (10:5). In July 1987, the AFLMC study of WRM vehicle Management Problems concluded with the completion of a functional description of a WRM Fleet Management system (AFLMC Report LT860840). AFLMC subsequently requested the Air Force Standard Systems Center

to develop a software program for use on a personal computer with the capability of extracting as much information as possible from existing automated systems.

The common thread that appears in all the inspections clearly identifies the central problem as a database management problem. The necessity of tracking up to 185 fields (an item of data) of information for 1200 vehicles on a daily basis is without question beyond the ability of manual tracking systems. Effective WRM vehicle management requires that this be done to meet the goals of the Air Force WRM management program. Determining the current posture of the fleet requires analysis of the database to determine shortfalls as an indicator of readiness. Such problems are ideally suited to a computer's ability to rapidly manipulate a large amount of data and provide specifically formatted information about its attributes. A management information system (MIS) of this type will provide the vehicle manager assistance in rapidly making day to day decisions without resorting to cumbersome manual methods. The goal of this system must be to free the fleet manager to perform the critical planning and analysis function and provide a rapid reporting capability during warplan activation. This computer based MIS must automatically reconcile the source document differences and provide base-level transportation personnel with consolidated reports that identify scheduled actions and summarize historical data specifically for WRM vehicles.

Research Questions

In order to develop a computer based MIS for WRM vehicle management that will meet the needs of base-level transportation personnel and resolve the problems identified in the 1984 and 1986 Air Force IG reports, the following research questions must be answered:

1. Can a computer-based MIS be developed that will meet the requirements identified in the AFLMC functional description (10:15-31)?
2. Can a computer-based MIS be developed that will make use of available technology and equipment?
3. What is the appropriate computer language to be used for system development and what format should the software take to ensure user-friendliness and yet maintain compatibility with existing systems?
4. Will the system be required to access currently existing databases or computer systems to obtain input?
5. What criterion must the validation and testing meet, to ensure successful implementation and acceptance by base-level transportation personnel?

Limitations

This thesis will not deal with any classified information nor will the prototype system be designed to provide any security measures that would be appropriate to systems that handle classified information. The information generated by the WRM management system in any summary reports or fleet posture analysis must be considered of a sensitive nature and must be appropriately limited to personnel with official purpose. Dispersal or staging information that might be input into the system is considered classified and appropriate security measures must be taken by fleet

management personnel. While the system may handle classified material it will be assumed that the way in which it accomplishes this task can be developed independent of security concerns. The proposed system will not be designed to reaccomplish actions that are accomplished by the AFLMC VAMS system. Additionally the proposed system will not duplicate tasks accomplished by the Air Force Standard Systems Center On Line Vehicle Integrated Management System (OLVIMS).

Thesis Organization

This research involves the development of a computer program to meet the objectives of the AFLMC functional description and resolve the current WRM vehicle management problems. Thus a literature review will be integrated with the methodology chapter due to its limited scope. The traditional findings and discussion will be followed by a chapter on program documentation, and the final chapter will be on Conclusions and Recommendations. Because of the unique aspects of this thesis the structure will be as follows:

Chapter I: Introduction
Chapter II: Methodology
Chapter III: Findings and Discussion
Chapter IV: Program Documentation
Chapter V: Conclusions and Recommendations
Appendix A: Program Coding
Appendix B: Glossary of Terms
Appendix C: User Guide
Bibliography

II. Methodology

Objectives

The primary objective of this research was to determine if a database management information system (DBMS) prototype could be designed and programmed within the following guidelines:

1. The system must be able to run on the hardware (see appendix B) currently available to most Air Force Transportation Squadrons.
2. The system must be capable of integrating with the software (see appendix B) currently being used to manage the peacetime fleet.

The system referred to in the guidelines actually has a double meaning. First, it clearly refers to the computer program and its collection of sub-modules as a "system." Secondly, and more important, it refers to a particular strategy that will be used to determine the operational parameters and structural subcomponents of the DBMS. A system consists of a set of objects and their relationships between the objects, operating within and subject to an environment (30:1-2). The "systems approach" requires a commitment focusing on the performance of the system in its intended environment, rather than concentrating on optimizing subcomponent performance. If the concentration were shifted to optimizing subcomponent performance, the overall system performance might not meet the desired operational goals. Within the realm of computer science, the systems approach is often implemented through "top-down design" (see appendix B)

but is somewhat distinct from this design approach by its concentration on operation within the overall environment. Viewing the development of a DBMS with a "systems approach" will ensure the effectiveness of the system while using "top-down" design will ensure the efficient operation of the computer program.

Definition of DBMS Requirements

A review of the AFLMC Functional Description for a WRM Vehicle Management System (AFLMC Report LT860840) revealed that it provided a thorough definition of DBMS requirements. Telephone contacts were established with the author of the report and the original AFLMC points of contact to ascertain if any changes in DBMS requirements had occurred since the AFLMC field visits. The results of these conversations clearly emphasized the need to develop a new DBMS rather than try to adapt existing programs. Additionally, the consensus of opinion favored the system description provided by the AFLMC report with little change. Thus, the current dating (July 1987) and excellent quality of the AFLMC functional description, obviated the necessity of conducting much field research to determine DBMS requirements.

The traditional library research centered upon a review of available library and commercial software users manuals, related theses, computer science texts, and current computer magazines. The requirement to develop a database management system implementation rather than adapt an existing one,

focused much of the literature research upon available software to determine which DBMS software package would be the most appropriate for system development.

Design and Development of DBMS

The prototype system was designed to operate on IBM™ personal computers or IBM™ compatible computers with a minimum of 640 kilobytes of random access memory (RAM) (see appendix B). This type of computer standard was chosen because it was determined through telephone contacts with USAFE and PACAF transportation representatives that all of the using bases had or would have access to an IBM™ compatible computer. The Air Force's choice of the Zenith™ 100 and 248 model computers as the Air Force standard personal computer and subsequent availability for purchase under Government Service Administration contract served to standardize the hardware this system would require.

Much of this system's output is in the form of printed reports. The printer standard chosen was the Epson™ or Epson™ compatible printer. This standard is currently widely emulated, and is present in those printers available through GSA contract.

The data storage medium currently being used consisted of hard disks and floppy disks (see appendix B) in two different configurations:

1. Systems with one or more floppy disk drives.
2. Systems with floppy disk drives and hard disk drives.

The common factor in both configurations was therefore chosen as the medium for which this system would be designed. This choice is reasonable because a system designed to store the program and data on floppy disks can be easily stored and operate without change on a hard disk drive while the reverse might not be true or practical.

The user's computer system may be configured with any computer monitor because the program is able to determine which monitor is present and adjust accordingly. If color or enhanced graphics adapter monitors are used, the screens will be in multicolor. If a monochrome monitor is used, the user will notice the characters and lines will be shaded differently.

The desired end product was a program that would fit on a 5 1/4 inch dual side floppy disk formatted to 360 kilobytes. It would remain resident in the random access memory (RAM) allowing the user to remove the program disk from the disk drive after loading was complete and insert the database disk. The program must therefore be completely self-contained in the computer's RAM and not make repeated function calls to the program disk to operate. This arrangement would allow the entire floppy disk to be used for database storage. The standard 5 1/4 inch dual side floppy disk has approximately 360 kilobytes (see appendix B) of storage capacity (26:3.74). The database search algorithms (see appendix B) were written to permit the insertion of more than one floppy disk allowing relatively large vehicle

databases to be maintained on several floppy disks. The user could control which vehicles data would reside on each floppy disk, permitting in theory any size vehicle database to be used. Thus, the system can quite easily manipulate the 1200 vehicle record maximum expected at any using base (10:15).

The "top-down" design process was chosen to ensure a logically complete system with a modular structure. A modular structure permits simplified error checking and easy modification (23:13). By functionally grouping tasks such as adding vehicles to a database, or editing vehicles in a database, common procedure modules were designed to perform these tasks regardless of the database in use. This technique eliminated needless duplication of code (see appendix B) thereby significantly reducing total program size while enhancing speed. Standardized task oriented modules also dramatically reduced the amount of code "debugging" (see appendix B) that was necessary.

Error trapping techniques were used in direct proportion to the severity of damage that would occur if erroneous data entered the database. The majority of error trapping procedures were confined to the data input modules. System quality control operates best when the quality of the data is controlled as it enters the system instead of correcting errors once they are already in the system (23:126-147).

The entry of data into information fields was rigidly controlled by pre-formatting what the user could enter.

Alphabetical characters are converted to upper-case as they are entered into the character fields by the program rather than the user. This eliminates the necessity of developing search algorithms capable of recognizing that an uppercase "A" is to be found and sorted in the same manner as a lower-case "a" despite the computers internal programming dictating that they are different.

Duplicate entries are not permitted and are controlled by searching the database for the vehicle registration number before it is added. If the record exists, it can be overwritten by the new record but both cannot exist in the database at the same time. Eliminating duplicate entries before they occur, prevents the ever "expanding database" phenomena and therefore reduces the amount of time the user must spend grooming the database to increase output accuracy. Blank records are also eliminated before they are permitted to occur. This greatly enhances the speed of search routines.

Databases are only opened long enough to permit the data to be placed in the fields and then are immediately closed. To illustrate this further, if a user adds 100 vehicle records to a database it will be opened and closed 100 times. This practice reduces the possibility of destruction of the database that could occur during sudden power outages or when the user accidentally shuts the computer off during data entry.

Multiple databases are used that are grouped by the source the information is derived from, or the type of information contained in the database (example: exercise or release case data), to enhance the ease of data entry. If a single database incorporating all 185 fields of information were used, a sudden loss of power could cause a complete loss of all data. Thus, using several smaller databases minimizes this possibility. The use of multiple databases also permits the user to add, edit, or delete data obtained from the same source document preventing the user from having to search several different source documents for one vehicle record. Several smaller databases can be searched much faster than a single database with exactly the same fields of information.

The only field of information that must be duplicated in every database is the vehicle registration number. This is logically sound because all vehicle management activities require the vehicle to be identified by its registration number as an integral part of the vehicle management function. This also permits linking the databases on this key field when a particular report or view screen draws data from several different databases.

The interface between user and computer makes use of functionally grouped menu driven options that were created with two primary goals in mind to ensure "user friendliness" (see appendix B):

1. Speed - How fast can the user move through the options to accomplish the desired task.

2. Ease of learning and use.

Menus were limited to eight options with the ability to exit to the main menu from any point of the program. Menus were chained together and limited to three levels deep from the main menu to maintain user friendliness (23:175-180). The user can only exit the system at the main menu. Thus, the user can exit the system at any level with only two keystrokes. The option to exit the menu or the system is presented as the last function, and the program automatically stores "0" to the choice prior to displaying the menu. This allows the user to just press the return key to exit, thus speeding the learning process.

Field Test

Testing consisted of pre-field and field testing. The pre-field testing occurred at all stages of program development. A database of all WRM vehicles assigned to PACAF was obtained and used periodically to test the ability of the system to sort, retrieve and manipulate data from a large database. Testing during module development reduced the degree of final system errors encountered during Field testing. A disk containing vehicle information extracted from the Air Force Standard Systems Center OLVIMS system was used to test the ability of the program to accept, sort, and retrieve data from another database. Both input sources test the ability of the system's automated data entry concept designed into the WRM vehicle management system.

The field testing phase consisted of submitting the software program to the AFLMC vehicle division to test the overall performance characteristics and how well the WRM system interfaces with the AFLMC Vehicle Automated Management System (VAMS). Additionally a copy of the WRM system was sent for field testing to the vehicle division of PACAF/HQ.

Feedback from all testing phases was used to modify the program with subsequent in-house retesting to ensure errors were eliminated.

Validation

The process of validation consists of verifying that the software meets the stated design objectives and performance specifications. Validation was conducted through the AFLMC vehicle division to ensure that the overall objectives of the AFLMC functional description were met.

III. Findings and Discussion

In Chapter I, five specific research questions were identified that must be answered in order to develop a computer based MIS for WRM vehicle management that will meet the needs of base-level transportation personnel and resolve the documented deficiencies identified in the 1984-86 Air Force IG reports.

System Requirement

Question I: Can a computer-based MIS be developed that will meet the requirements identified in the AFLMC functional description?

The AFLMC Functional Description For A WRM Fleet Management System groups the system requirements into three general areas:

1. Specific Performance requirements.
2. Response time requirements.
3. System functions requirements.

Many of the requirements identified in these three areas are functionally identical to the requirements that existed for management of the peacetime fleet prior to the development of the Vehicle Automated Management System (VAMS). The VAMS system was developed with the same basic requirements that the WRM system must satisfy. The VAMS system could contain more data than the WRM system because it is designed to accomplish the authorization/assignment function for both vehicle fleets. Thus, the amount of data that each of the systems must manipulate are comparable in size. The requirements for data

accuracy, and validity are identical since in most cases the exact same fields of information used by the VAMS system must also be used in the WRM management system to generate its reports and view screens. The VAMS system was written using the dBase II™ database language. Two more powerful versions of dBase have been introduced since the VAMS system was written and a third was just announced. Thus, because the requirements for both systems are the same, it is reasonable to assume that a prototype system could be developed to satisfy the AFLMC systems requirements.

The major difference between the VAMS system and the proposed WRM system is the way in which each system derives its input. The VAMS systems receives its information from three sources, the Vehicle Integrated Management System (VIMS), the Vehicle Authorization Listing (VAL), and through manual updates. The WRM system receives its input from the VAMS system, the VIMS system, and through manual update.

This difference becomes important because the prototype development language must be able to read a DBASE II™ file structure and convert it into a file that it can use. By programming the WRM Management System to read VAMS data files, the user can take advantage of the monthly data reconciliation that has already occurred in the VAMS system thereby greatly increasing data accuracy and reducing manual input.

The prototype system should also be able to access the data files created by the VIMS system directly in the same manner that the VAMS system does. Programming the WRM system to read VIMS created files will enhance its versatility and permit an alternate method for receiving automated input. Figure 1 depicts how the WRM Vehicle Management System uses the VAMS and VIMS output as its input sources.

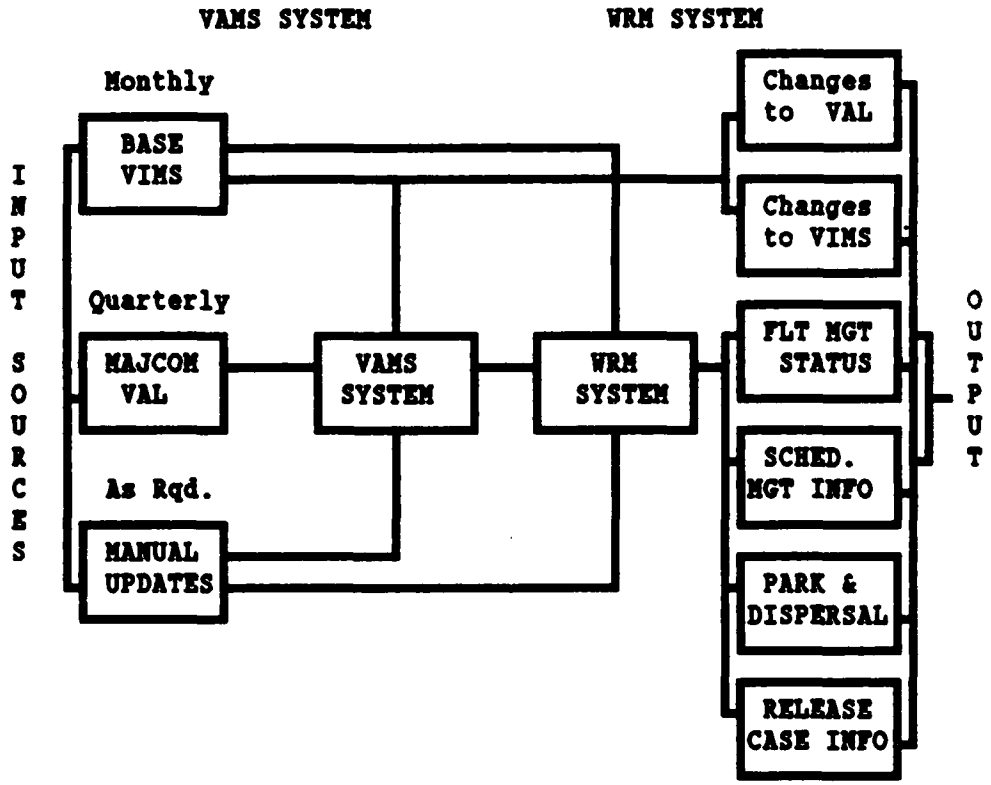


Fig. 1. Prototype System (10:12)

If the direct VIMS to WRM data transfer option is chosen, the user assumes a greater burden to reconcile and validate data accuracy than if the VAMS to WRM data transfer option is chosen. The latter option provides a data source that has

already been reconciled and validated. Permitting both automated data input methods in addition to providing manual input ability, enhances the WRM system's capability to interface with its operating environment and yet decreases its dependence on any single source.

The unique programming requirements eventually became a driving force behind the choice of the prototype development language. Further discussion on the specific development language chosen and how the automated data transfer requirements were met is explained in research question three in this chapter. Thus, all requirements outlined in the AFLMC functional description were met by the prototype WRM management system.

Use of Available Technology and Equipment

Question 2: Can a computer-based MIS be developed that will make use of available technology and equipment?

The most important aspect of this question is the degree of hardware standardization that exists among all the using bases. Currently all using bases have or have funded orders for Zenith Z248™ computers configured with one floppy disk drive, and a twenty megabyte hard disk drive. This was confirmed during telephone contacts with USAFE and PACAF Headquarters transportation representatives when the DBMS requirements were initially defined. The Air Force standard GSA printers are capable of emulating an Epson™ printer and are either already present at the bases that have computer

systems, or have been ordered with the computer system. The relative standardization of hardware at using installations greatly reduced the potential problems associated with possible software to hardware incompatibility. Virtually all of the potential prototype development languages were capable of developing programs that run on IBM™ compatible computers and Epson™ compatible printers.

The other aspect of this research question that had bearing upon the eventual choice of the development language dealt with the possibility that the natural limitations of the hardware and commercially available software might not permit development of a powerful enough system to cope with the requirements placed upon it. This question was satisfactorily answered during the literature review that predicated the eventual choice of the prototype development language.

Software Identification

Question 3: What is the appropriate computer language to be used for system development, and what format should the software take to ensure user-friendliness and yet maintain compatibility with existing systems?

Three basic types of software packages were considered in the search for the prototype development language, database management software, programming languages, and integrated packages.

Several important factors were identified as necessary attributes that the prototype development language must possess:

1. The prototype language must provide a range of programming features that permit modular development.

2. The prototype language must permit complete programming control of the user interface design to ensure the final system exhibits user-friendliness.

3. The prototype language must be flexible and powerful enough to meet the performance specifications described in research question one.

4. The prototype language must be able to be programmed to interface with the VIMS data files and with the dBase II files created by the VAMS system.

5. The prototype language must be available through Air Force standard supply channels.

6. The prototype language must be able to be compiled (see appendix B) to enhance its speed, but more important to provide standalone capability after system development is complete.

Integrated packages are actually a system containing a word processing module, a spreadsheet module, and a database manager. The key strength of an integrated package is its ability to pass information between the modules allowing the user a vast range of choices for displaying and manipulating data. The main weakness of an integrated package is the reduced capability of each module. This reduced capability occurs as a result of the tradeoff that must be made to permit integration between the modules. The integrated packages this research examined provided only limited programming features and none of the packages had any commercial compilers available to produce stand alone applications. Additionally, the major feature of the integrated package that would be used by the WRM system would be the database portion. Word processing software and spreadsheets were already available at

the using bases. Thus, even though the integrated concept has merit, the "state of the art" software packages currently available are neither powerful enough to handle this application nor are they capable of being compiled. These limitations eliminated integrated packages for use as the prototype development language.

Programming languages (BASIC, PASCAL, FORTRAN etc) certainly provide the flexibility and power necessary to develop virtually any system application and commercially available compilers abound. Each language examined had certain characteristics that made it better suited to develop specific types of applications. The "C" language was initially created to be used as a database development language. This is supported by the number of database management languages that were developed using the "C" language (dBase, rBase etc). Thus it would appear that perhaps the "C" language might be the logical choice for the prototype development language. The major drawback in adopting this approach was the amount of programming time that would be necessary in the initial phase of system development. It seemed more logical to turn to a database management language that contained tried and tested algorithms for manipulating data rather than try to develop them for this thesis. Thus, despite the sacrifice in power and flexibility associated with using a "higher level" (see appendix B) language, the choice was made in favor of a database management language to shorten development time.

Despite the abundant number of database management software packages available, a clear cut division existed between those packages with programming languages and those without. "Database systems capable of dealing with large amounts of data generally fall into three categories: hierarchial, network, and relational (27:93)." Hierarchial databases are organized into distinct levels with the tree like structure readily apparent to the user as commands are entered to query the database. Network systems use mathematical set theory establishing ownership and membership properties to groups of data. A series of common linking relationships are established on common set membership and subsequently used to search for data. Relational databases allow access to multiple database files as long as those databases are related through common fields of information. An illustration of this might be the use of a Social Security Number field (see appendix B) of information as the common tie between a database that might contain financial information and a database that might contain medical information. Programmable database languages can be further classified as procedural or non-procedural. The key difference between the two types is that non-procedural languages allow the programmer to tell the computer what is to be done while the procedural language allows the programmer to tell how it is to be done (29:94).

The database management language that was chosen as the prototype development language was dBase III Plus™. The key

features of this programming package that made this choice the logical one are:

1. Available on GSA contract through standard Air Force supply channels.

2. High degree of compatibility between dBase II™ and dBase III Plus™ programs and database files.

3. Highly programmable containing the best features of a hierarchial database with relational capabilities allowing the programmer to completely design the user interface.

4. Commercial availability of compilers to produce standalone programs, thus eliminating the requirement to purchase a copy of dBase III™plus at each using installation.

5. dBase III Plus™ allows the use of procedure files permitting commonly used standard modules to be developed greatly reducing the overall amount of code in the program.

Interface with Existing Systems

Question 4: Will the system be required to access currently existing databases or computer systems to obtain input?

The WRM vehicle management system can function without automatic data input from existing systems. This ability to operate independent of other systems is desirable to provide flexibility. This system can operate on a portable computer in the field and still provide the WRM fleet manager invaluable assistance. This research determined that the WRM system should be developed to access both the VIMS and VAMS system output. This will ensure reconciliation of data between all three systems increasing the overall validity and

accuracy of the data. Additionally a substantial amount of user input time can be reduced if the system is designed to automatically receive VAMS and VIMS input.

Validation and Testing

Question 5: What criterion must the validation and testing meet, to ensure successful implementation and acceptance by base level transportation personnel?

The performance requirements outlined in the AFLMC functional description can best be met by repeated pre-field testing as program development occurs, followed by a period of field testing monitored by AFLMC representatives. Since this is a prototype system it is expected that some changes will be necessary as the system is fielded and user responses are accumulated. The concept that must overshadow all validation and testing criterion is "user friendliness." This concept must prevail throughout the program development phase to ensure that base-level transportation personnel will find the system easy to learn and use, but still effective. The development of a sound user's manual that can provide the new computer user with sufficient support to encourage rather than discourage repeat usage is crucial to the successful acceptance and implementation of this system. Thus, designing the human factors in at each stage, coupled with extensive pre-field testing will ensure that the user can use the system effectively for what it was designed to accomplish.

IV. Program Documentation

Program Organization

The WRM Vehicle Management program code was written using Ashton Tate's dBase III PLUS™ database management software. A complete list of program code is included in Appendix A. The modular structure and organization of the WRM Vehicle Management System is depicted in Figure 2.

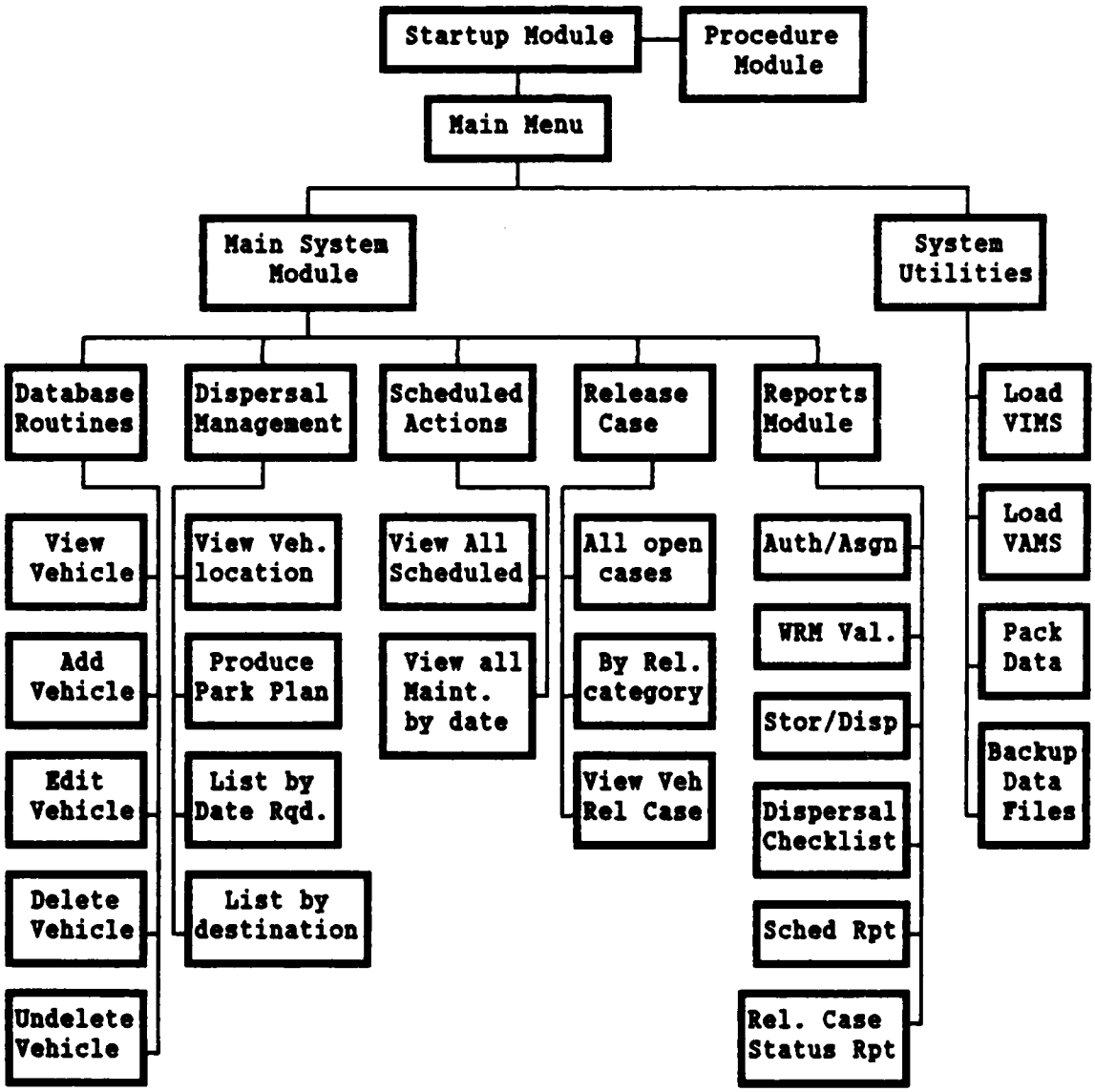


Fig. 2. Organization of WRM Management System

The modular structure of the WRM Management System is grouped by common function to take advantage of standardized procedures that are loaded into RAM when the user starts the program. The procedure module remains RAM resident and is used by every sub-module of the program. The database routines module actually accesses 16 different databases that vary in structure, size, and purpose. Each of the databases are indexed on specific fields of information to facilitate rapid search techniques. An index file for a database contains record numbers and a minimum of one field of data. The index file is sorted in the order corresponding to the key field of data it contains. The search for a specific vehicle record in a database is accomplished by searching the index file and then matching the index record number to the database record number. This technique dramatically increases the speed of the search algorithm and also eliminates the time consuming constant sorting of the database.

The dispersal/distribution, scheduled actions, and reports modules permit the user to view and print a variety of different arrangements of the 80 different fields of information on each vehicle. The user can access any of the databases and view or print information on a single vehicle or vehicles grouped by a common characteristic like management code, or maintenance status.

The utilities module provides the means by which the databases are automatically updated from the Vehicle Asset Management System and the Vehicle Integrated Management

System. The pack sub-module permanently deletes vehicle records that were previously marked for deletion from all databases. This is significantly different than the delete/Undelete function of the database routines module, because records are only moved from the database in use to a temporary database when deleted and returned from the temporary database when undeleted.

Introduction

When the user begins the program by entering "WRM" and pressing return, a message appears advising that the program is being loaded. Once the procedure file is loaded into RAM the user is greeted by the rights and warranties screen depicted in figure 3.

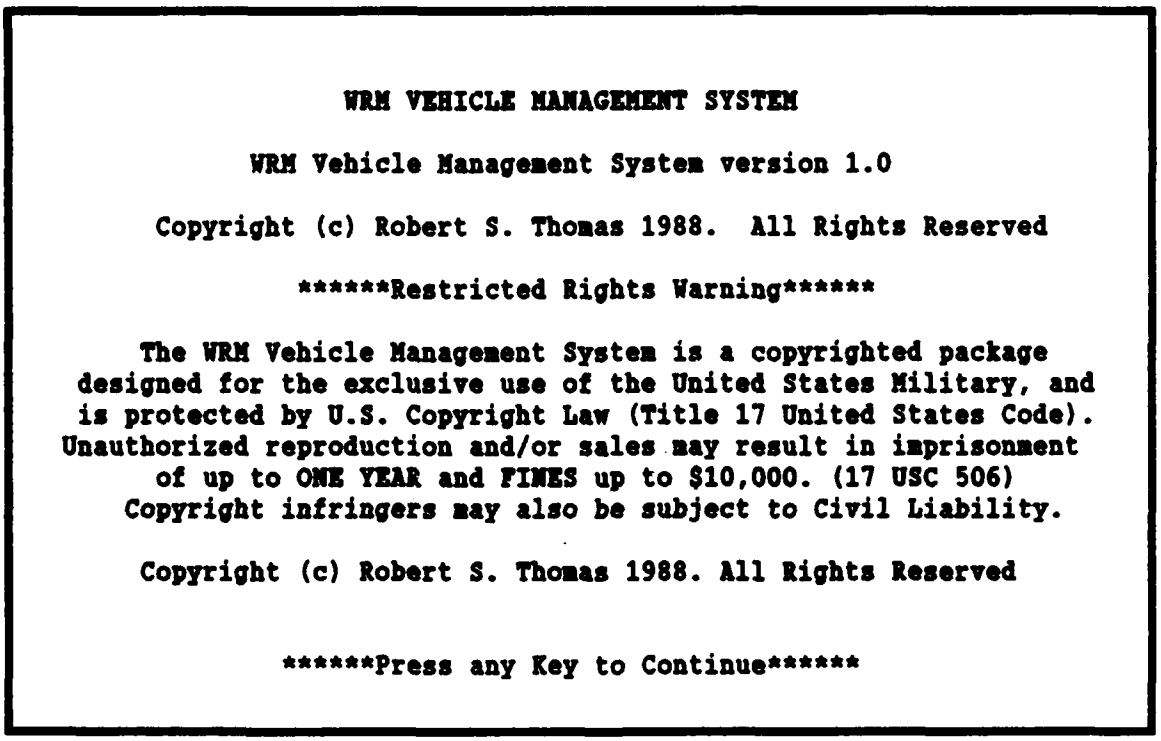


Fig. 3. Restricted Rights Warning Screen

The primary reason for restricting distribution rights is to prevent commercialization of the code outside of military channels. After the user presses return the startup menu is presented as shown in figure 4.

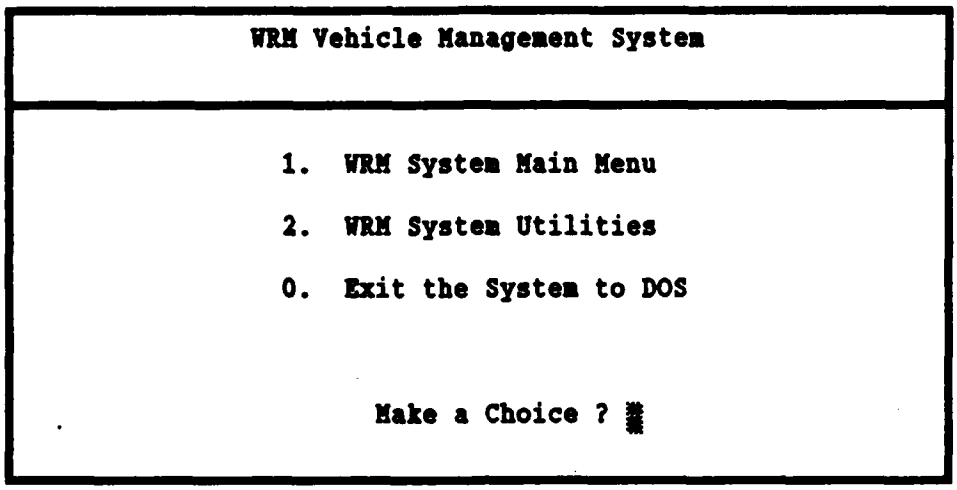
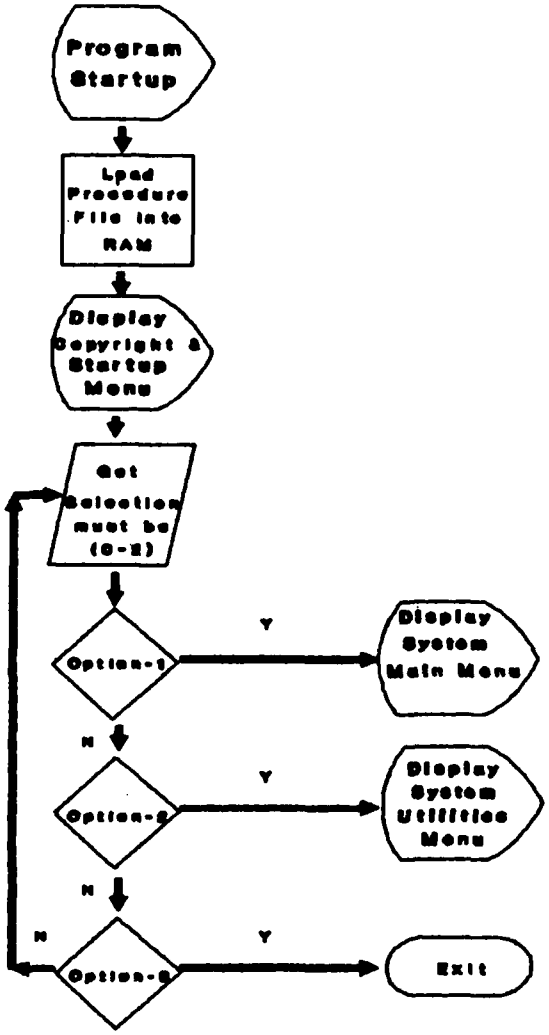


Fig. 4. Startup Menu

The startup menu offers the user the opportunity to enter the main body of the WRM system where all database oriented operations occur (add a vehicle etc.) or enter the utilities function where system level operations occur such as receiving input from other systems (VAMS, VIMS etc.). The darkened block marks the spot where the user enters the choice and is represented as it actually appears on the computer screen with the exception that choice "0" is already entered in the block. This same pattern is repeated throughout each module to facilitate rapid learning and permit the user to exit the system by just pressing return twice. Figure 5 depicts the flow of operations through the startup module.

Fig. 5. Startup.prg flow chart



Main Menu

Upon entering choice "1" the user is presented with the menu shown in figure 6.

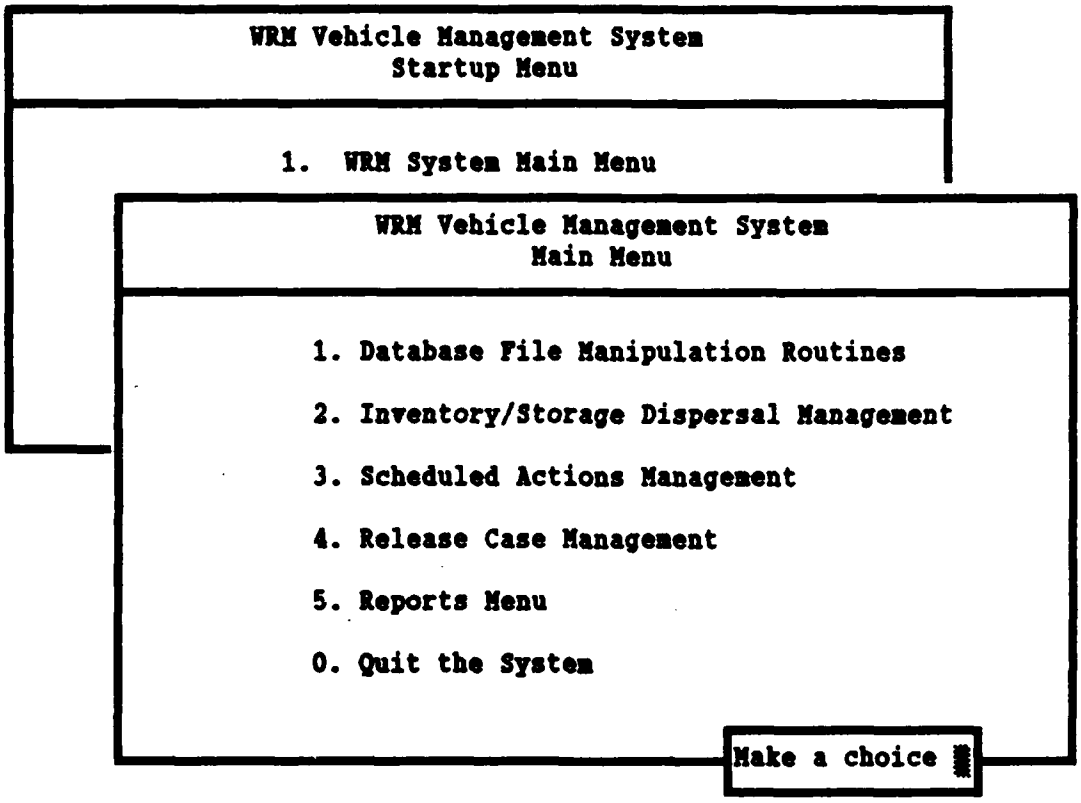
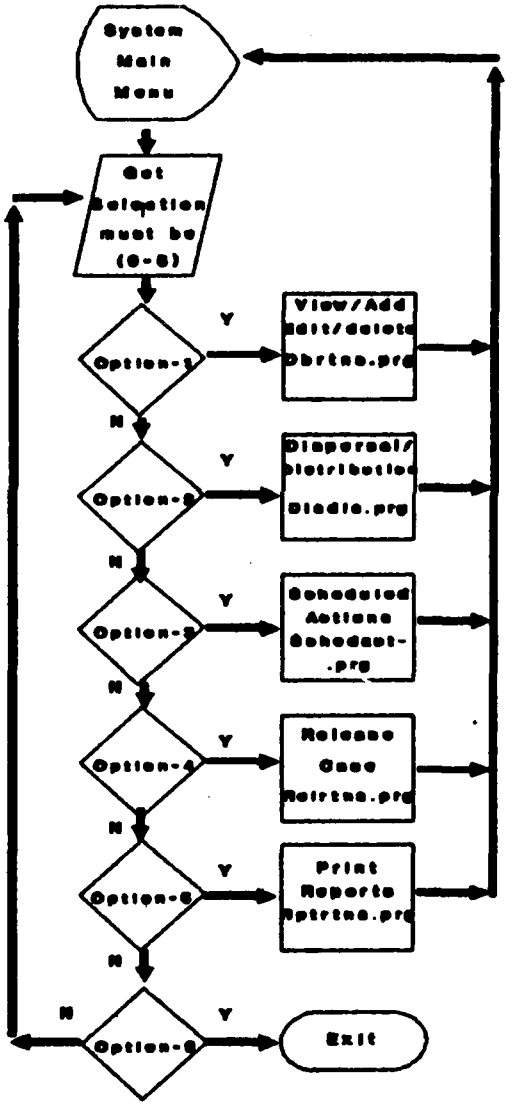


Fig. 6. WRM Main Menu Screen

At this point the user has entered the database operations oriented portion of the program. Once again "0" is stored to the choice box permitting the user to exit the system with one key-stroke. With the exception of choice "0" to exit the system, all other menu choices will return to the WRM Main Menu as the execution of the sub-module is complete and the user elects not to repeat the current operation. The flow of operation through the main menu to all sub-module options is depicted in figure 7.

Fig. 7. Main.prg flow chart



Database File Manipulation Routines

Despite the automated data update function available through the system utilities menu, the user may manually view, add, edit, delete, or undelete complete vehicle records or any field of information contained in a vehicle record. This permits the user to maintain complete control over all data stored within the WRM Management System. Upon entering choice "1" on the Main Menu the user is presented the view screen shown in figure 8.

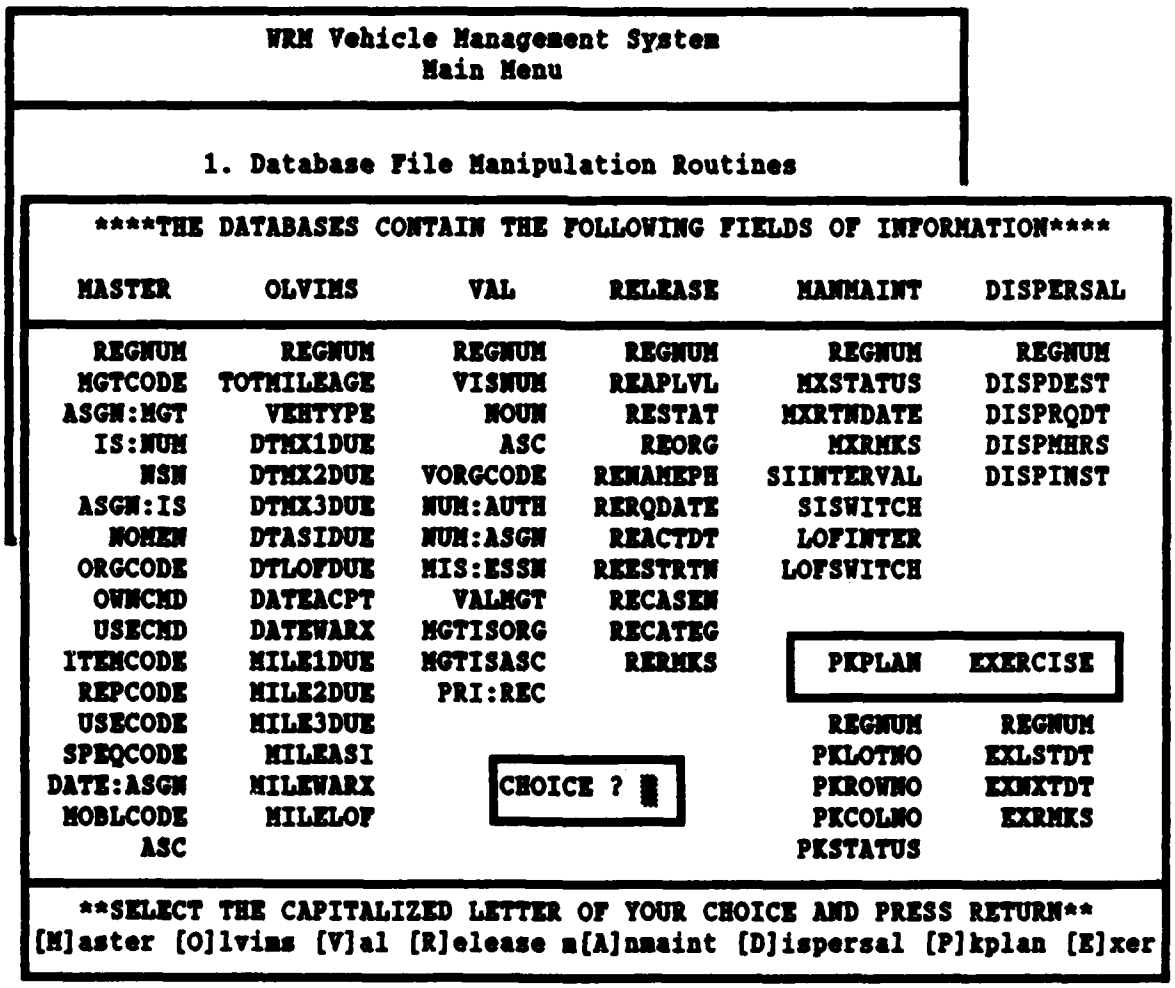


Fig. 8. Database Routines Menu

The fields of information in each database are provided in the Database Routines Menu to aid the user in choosing the correct database. When the user selects the database that is to be used, the view screen changes to provide a summary of its contents and the range of actions that can be performed upon it. Figure 9 depicts the view screen presented to the user.

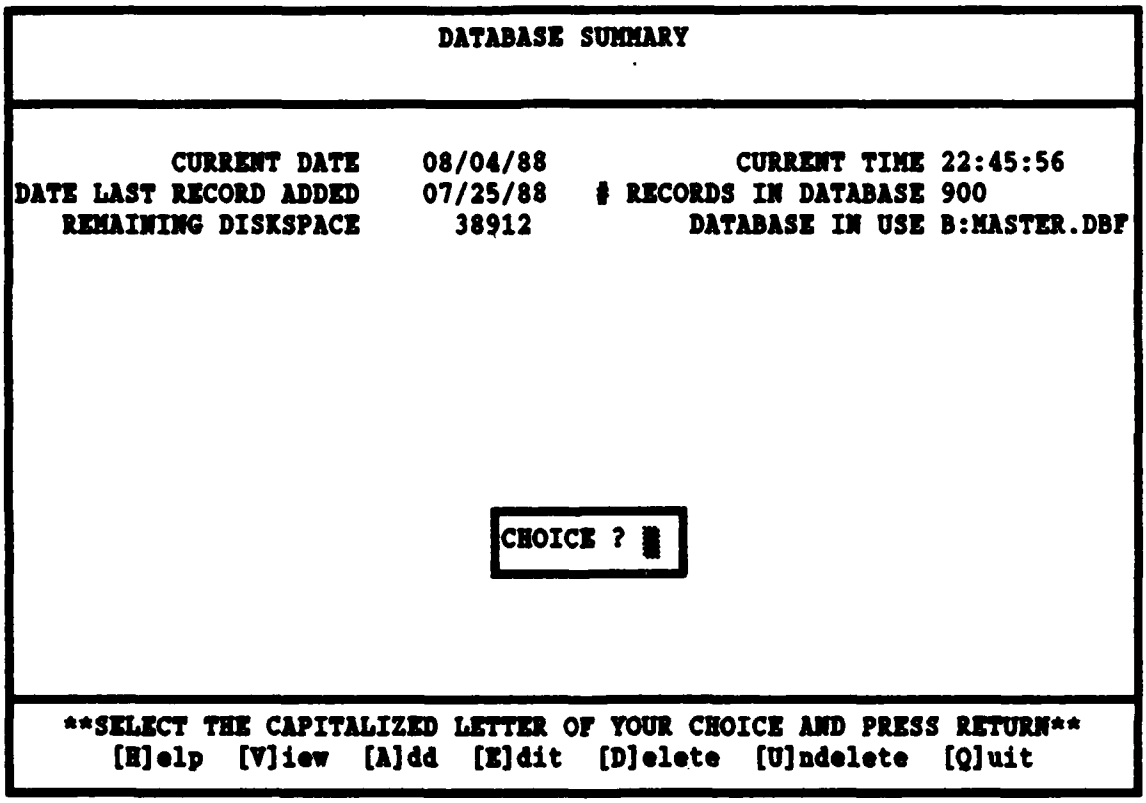


Fig. 9. Database Summary/Actions Menu

The database summary/actions screen is the same for all eight databases. "Q" is already stored to the choice block to permit the user to return to the main menu with one key-stroke. The database summary screen presents the user with important information about the database in use. The database summary information is presented at this stage to prevent the

user from attempting to add data to a full disk. A range of options are presented at the bottom of the screen. The view, add, and edit screens are formatted identically for the same database but vary between databases. The add/edit screens allow the user unlimited cursor movement within the darkened blocks. Once the final field of data is entered the user is required to press return before the data is added to the database. Each of the data entry blocks are formatted to permit the user to enter only the correct type of data. All alphabetical characters are forced to uppercase by the program to enhance "user-friendliness". To illustrate further, when the user is requested to input the vehicle registration number, numeric data is permitted in all eight of the provided spaces except space three where uppercase alphabetical data is required. The formatted fields validate the correct form of the data before it enters the database.

When the user presses return, a message appears at the bottom of the screen asking if the data is correct. If the data needs further editing , an "N" response will cause the program to loop to permit further editing. If a "Y" is entered the program searches the database for the registration number to determine if a record for this vehicle already exists. If a record already exists, the user is advised of this and permitted to overwrite the current record or abort this attempt. This prevents duplicate records from entering the database. If a "Y" is entered, the data will be added to the database and a message will appear requesting if the user

would like to add or edit more vehicle records. A "Y" response will cause the program to loop, while an "N" response will return the user to the main menu. Figures 10–17 show the different view/add/edit screens for each of the databases.

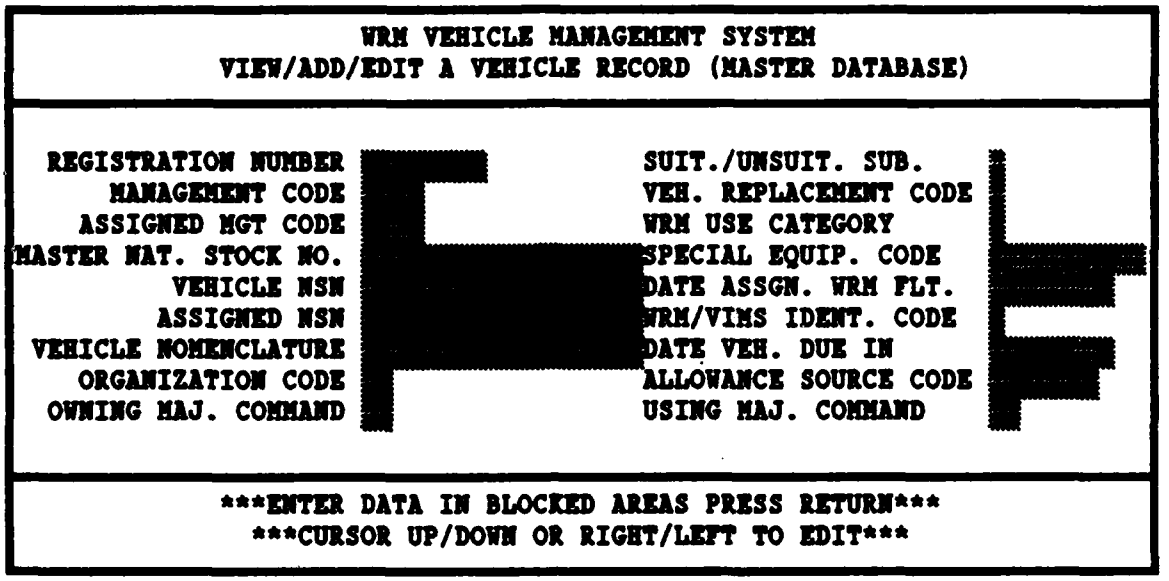


Fig. 10. Master.dbf View/Add/Edit Screen

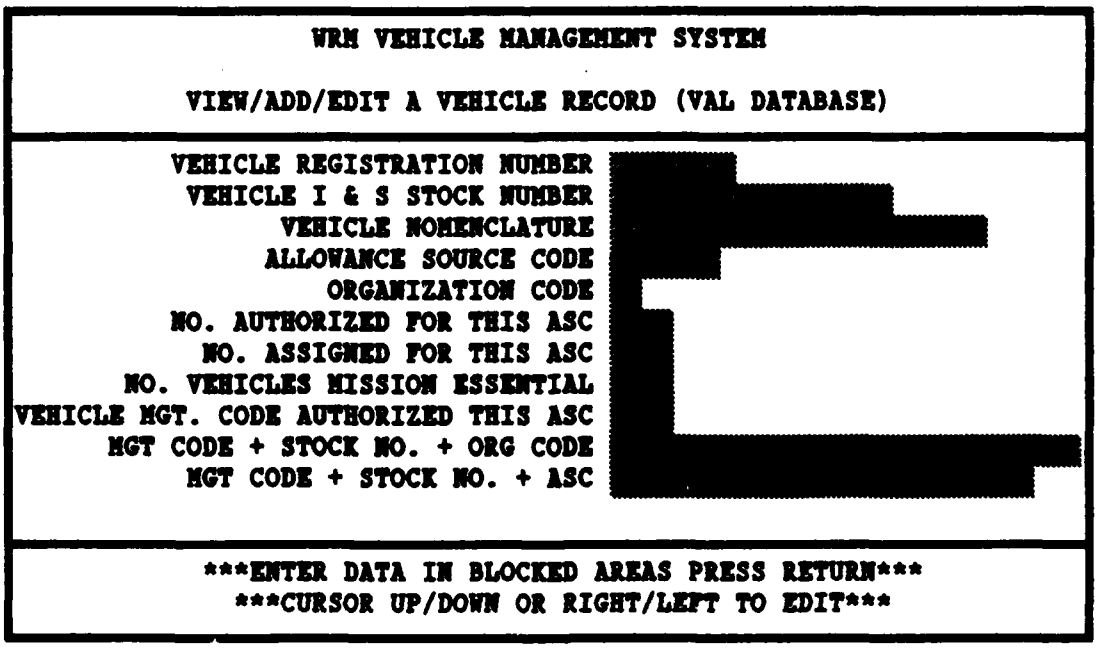


Fig. 11. VAL.DBF View/Add/Edit Screen

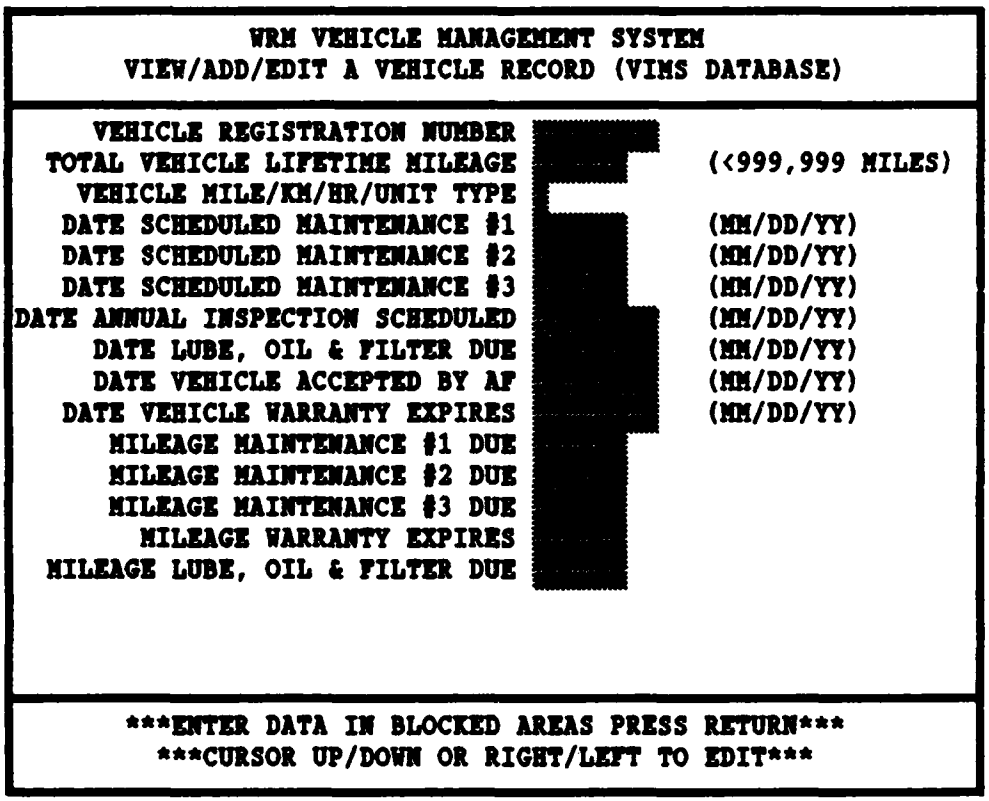


Fig. 12. VIMS.DBF View/Add/Edit Screen

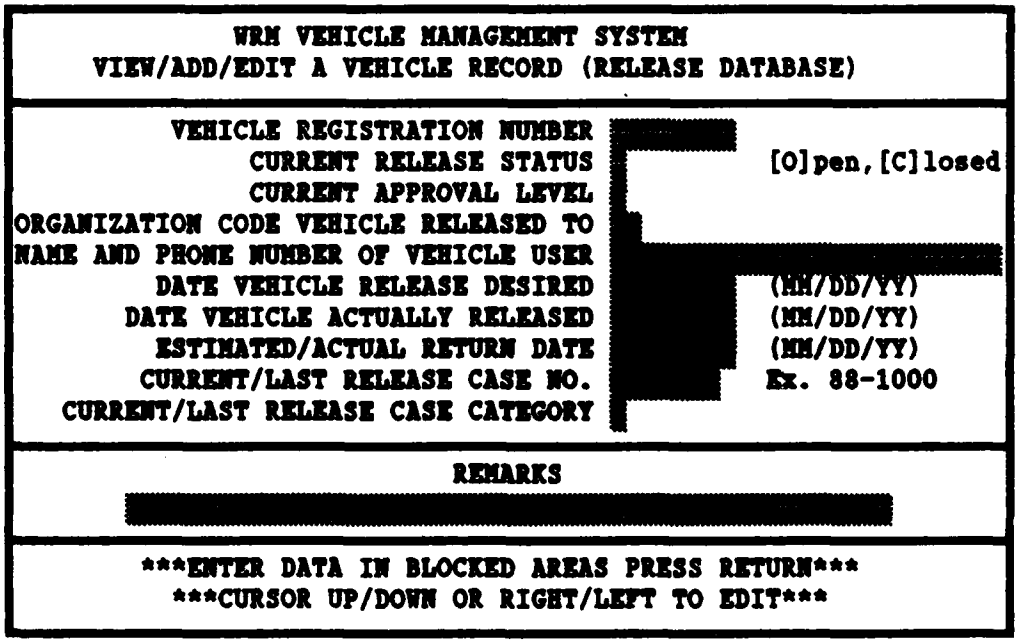


Fig. 13. RELEASE.DBF View/Add/Edit Screen

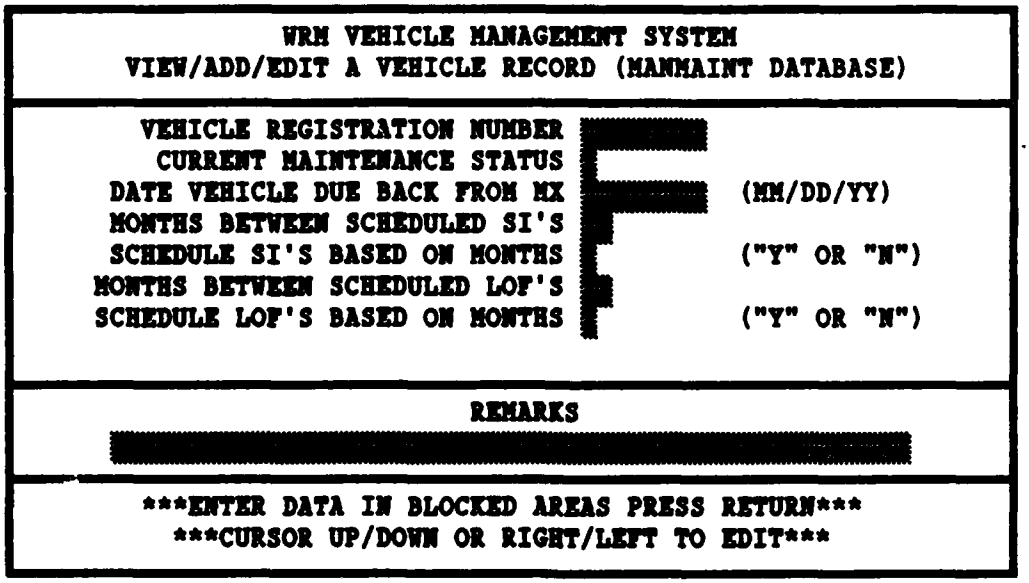


Fig. 14. MANMAINT.DBF View/Add/Edit Screen

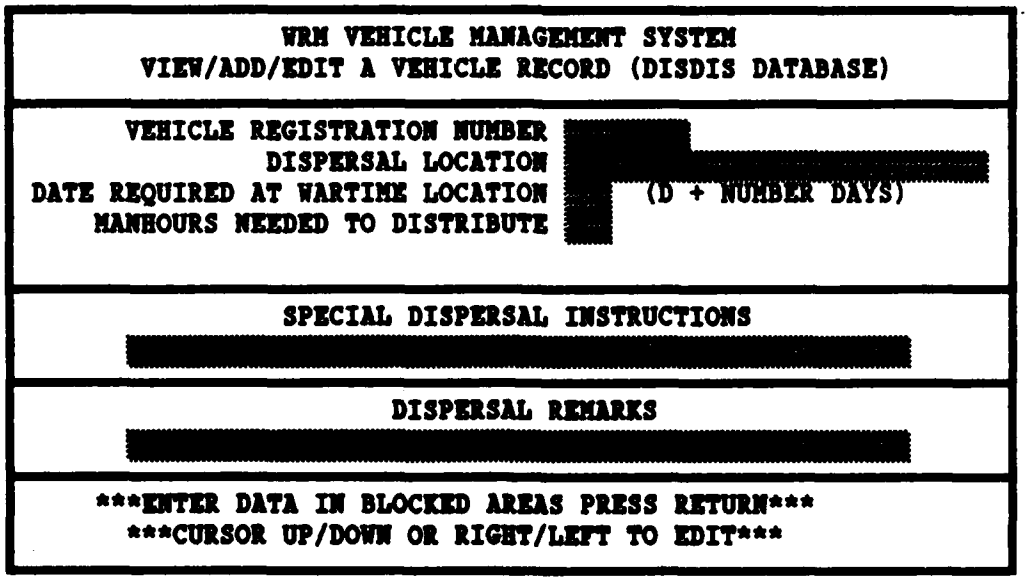


Fig. 15. DISDIS.DBF View/Add/Edit Screen

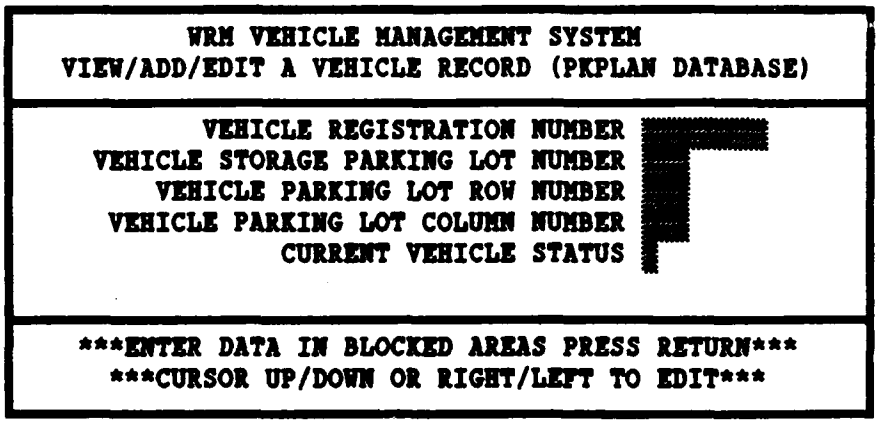


Fig. 16. PKPLAN.DBF View/Add/Edit Screen

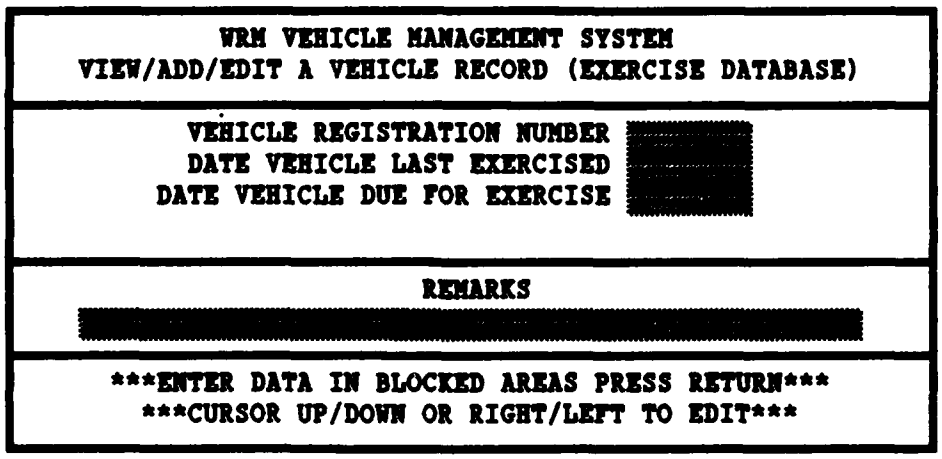


Fig. 17. EXERCISE.DBF View/Add/Edit Screen

With the exception of the master and vims databases which receive automatic and manual input, all other databases receive only manual input through the menus depicted in figures 11-17. If the user's computer is configured with a color monitor, the screen will appear with a blue background and all characters will be white. As the user moves from data field to data field the colors will change from white characters in a blue box to black characters in a white box, and back to the original colors after data entry is complete.

On a monochrome monitor the data will appear in reverse video image as it is being entered.

Each of the eight main databases has a duplicate temporary database that is identical in structure. The delete and undelete options uses all 16 databases, to enable the user to recover from an accidental record deletion. When a record is chosen for deletion it is deleted from the current database only after it has been copied to the duplicate temporary database. This prevents the data from being permanently lost. In order to permanently erase the data from the temporary database, the user must enter the systems utilities option and pack the databases. This choice will open all temporary databases and main databases and erase all records marked for deletion. The magnitude of the effect associated with packing the databases made it desirable to separate this action from normal day to day operations to prevent accidental packing. By removing deleted records from the main databases, search algorithms work faster, because they do not have to waste processing time deciding to ignore each deleted record as they are encountered.

The basic operation of the delete and undelete options makes use of the same flow of operations described for the view/add/edit processes. The user enters the vehicle registration number of the vehicle to delete or undelete. The view screens depicted in figures 11-17 portrays the data with a message at the bottom asking if this is the correct record. An "N" response will terminate the operation and the user will be

asked if they desire to search for another vehicle record to delete/undelete. If the user chooses not to repeat the operation, the program will loop back to the system main menu.

When the help option presented in each of the database summary screens is selected, a detailed definition of each of the fields of information is provided in addition to specific information about the database structure. The help screens are not designed to be the definitive answer to all problems encountered, but will provide basic information pertinent to commonly encountered questions. Figure 18 depicts one of the help screens for the master database.

DATABASE HELP SCREEN
FOR MASTER.DBF

The Master database contains the following fields information:

	Field name	Data type	width	decimal
[1]	REGNUM	CHARACTER	8	0
[2]	MGTCODE	CHARACTER	4	0
[3]	ASGNMGT	CHARACTER	4	0
[4]	ISNUM	CHARACTER	18	0
[5]	NSN	CHARACTER	18	0
[6]	ASGNIS	CHARACTER	18	0
[7]	NOMEN	CHARACTER	25	0
[8]	ORGCODE	CHARACTER	2	0
[9]	OWNCMD	CHARACTER	2	0
[10]	USECMD	CHARACTER	2	0
[11]	ITEMCODE	CHARACTER	1	0
[12]	REPCODE	CHARACTER	1	0
[13]	USECODE	CHARACTER	1	0
[14]	SPEQCODE	CHARACTER	10	0
[15]	DATEASGN	DATE	8	MM/DD/YY

****PRESS ANY KEY TO CONTINUE*****
****THE HELP FILE CONTAINS MORE INFORMATION****

Fig. 18. MASTER.DBF Help Screen

Dispersal Management

The dispersal/distribution module uses the pkplan and disdis databases to produce automated parking plans, and a variety of view screens that present the data fields in different formats according to the needs of the user. The user enters the dispersal/distribution module by choosing "2" from the System Main menu as is portrayed in figure 19.

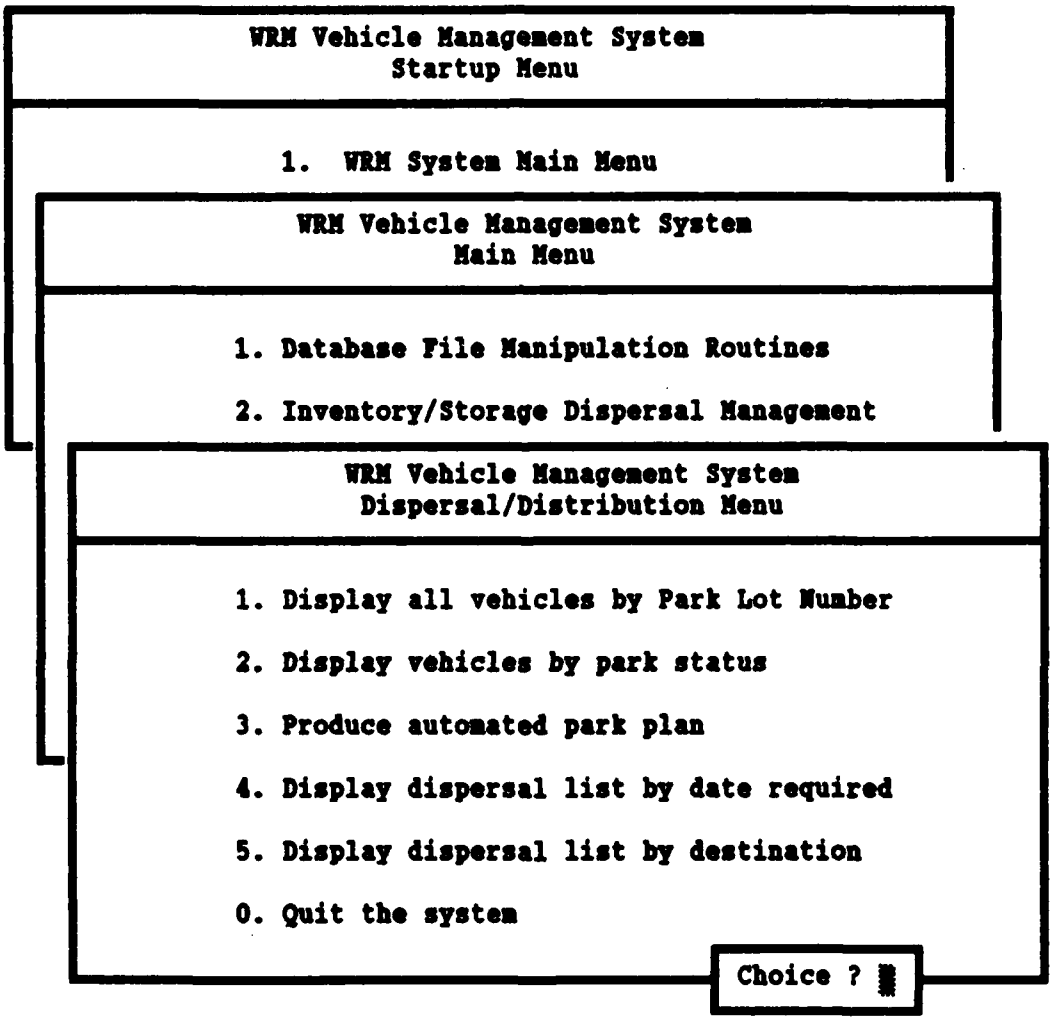


Fig. 19. Dispersal/Distribution Menu

The distribution portion of the dispersal/distribution module provides the option to display to the screen or print vehicles by location or parking status. If option "1" is selected the user is queried for the parking lot number and the output is directed to the screen or the printer. If the list is displayed to the screen, 20 vehicles and their locations will be listed and the user is instructed to press any key to continue. This sequence will continue until all vehicles in the selected parking lot have been listed. The user may print a portion of the entire list by pressing the shift key and the print screen key simultaneously and the contents of the screen will be routed to the printer. The entire list of vehicles in the selected parking lot is printed if the print option is chosen. Figure 20 portrays the output that appears on the screen.

Record #	Regnum	Pklotno	Pkrowno	Pkcolno	Pkstatus
1	88B10001	1	1	4	P
2	88B10005	1	1	6	P
3	88B10007	1	1	7	M
4	88B10009	1	1	10	P
5	88B10010	1	2	1	M
7	88B05942	1	5	3	P

There are no more vehicles in parking lot 1
Press any key to continue

Fig. 20. Parking lot display screen.

If option "2" is selected all vehicles are listed by parking status. Vehicles are categorized as "P" for (parked), "T" for (TDY), "M" for (maintenance), "O" for (other), and "S" for (storage). The range of categories is not limited to those previously stated. Any category may be used and all vehicles meeting this parking status will be displayed upon request. If other categories are used, they will not be recognized when the park status summary is displayed or printed. The summary screen will display the total number of vehicles and the number of vehicles in each parking status. The summary is displayed automatically after the last vehicles meeting the requested parking status have been displayed. The parking status screen is identical to the parking lot screen depicted in figure 20 with the exception that only vehicles of the requested parking status will be displayed. Option "2" permits the user to quickly ascertain the condition of the fleet and indirectly provides a measure of fleet readiness by identifying which vehicles are ready for immediate use by their parking status, registration number, and parking location.

If option "3" of the dispersal/distribution is selected, an automated parking plan will be printed. The automated parking plan is designed to print a parking lot of 200 vehicles. If there are less vehicles in the parking lot, the automated plan will still print a parking lot with dimensions of 10 columns by 20 rows (see figure 21), however the correct vehicles will be displayed in the correct parking spots.

PARKING LOT 1

ROW	COLUMN 1	2	3	4	5	6	7	8	9	10
1	88B10000 P	88B22000 P	78B32966 H	79B32967 S	86B41234 S	85B14100 P	81B38953 O	82B84213 H	75B12453 S	88B29887 P
2	82B22300 S	80B68310 P	85B41986 S	79B32968 H	81B01914 P	88B89312 P	76B81664 H	87B21459 H	83B00014 P	86B09701 H
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										

Fig. 21. Automated Parking Plan

Figure 21 displays 16 of the 20 rows that is actually printed when the user selects this option. This was done to maintain the margins for this thesis. If the parking lot is larger than 200 parking spots, the user can divide larger parking lots into 200 vehicle increments that match this automated parking lot program. The vehicle registration number and parking status are placed in each of the parking squares. The speed advantage of producing automated parking plans versus continually reproducing manual parking plans as changes occur, far outweighs the disadvantages associated with renumbering existing lots to within the 200 vehicle limit. When the automated parking plan is combined with the printed parking lot list (option 1), managing a high level of vehicle throughput during an exercise or actual dispersal can be realistically achieved.

The dispersal portion of the dispersal/distribution module begins with option "4" that provides the ability to display or print a dispersal list by date required. The date format used in this module is significantly different than the dates used throughout the rest of the WRM Vehicle Management System. The conventional dating method used during a dispersal or exercise is based upon the D + (number of days) format. This permits the relative dates that vehicles must be moved to be entered during the planning stages without actual dates being necessary.

When the user enters choice "4", a prompt appears requesting a beginning date and an ending date. This permits

the user to display any subset of the vehicle fleet that is dedicated to a dispersal or exercise. Thus, separate lists for any range of dates or a single date can be printed or viewed. This function allows the user to keep pace with changing conditions without having to view or print data for the whole vehicle fleet.

The dispersal module makes use of information that could be classified, therefore the user is advised of this prior to initiation of data entry or retrieval procedures. As mentioned in chapter one, no attempt has been made to encrypt the data contained in this database, nor has any password protection scheme been programmed to restrict user entry. The geographical dispersal destination codes and dispersal dates can be best controlled by adhering to already established security procedures, and by securing the personnel and equipment that use the classified information. Additionally the user must ensure that the computer being used is "tempest certified" for handling classified information. An acceptable possibility might be to keep a separate copy of the disdis database with the dispersal information already entered, in the LGX (Base Plans & Programs Office) or the LGTX (Transportation Plans & Programs Office) classified safe. This would also serve to enhance coordination between the WRM vehicle personnel and these offices. Thus, this portion of the module may see less routine use, but can be extremely helpful during dispersals or exercises.

Options "4" and "5" display and print the data in the same format and only differ by what subset of the database is displayed. Figures 22 and 23 depict the classified warning message and the display screen for displaying a list by d+ date required.

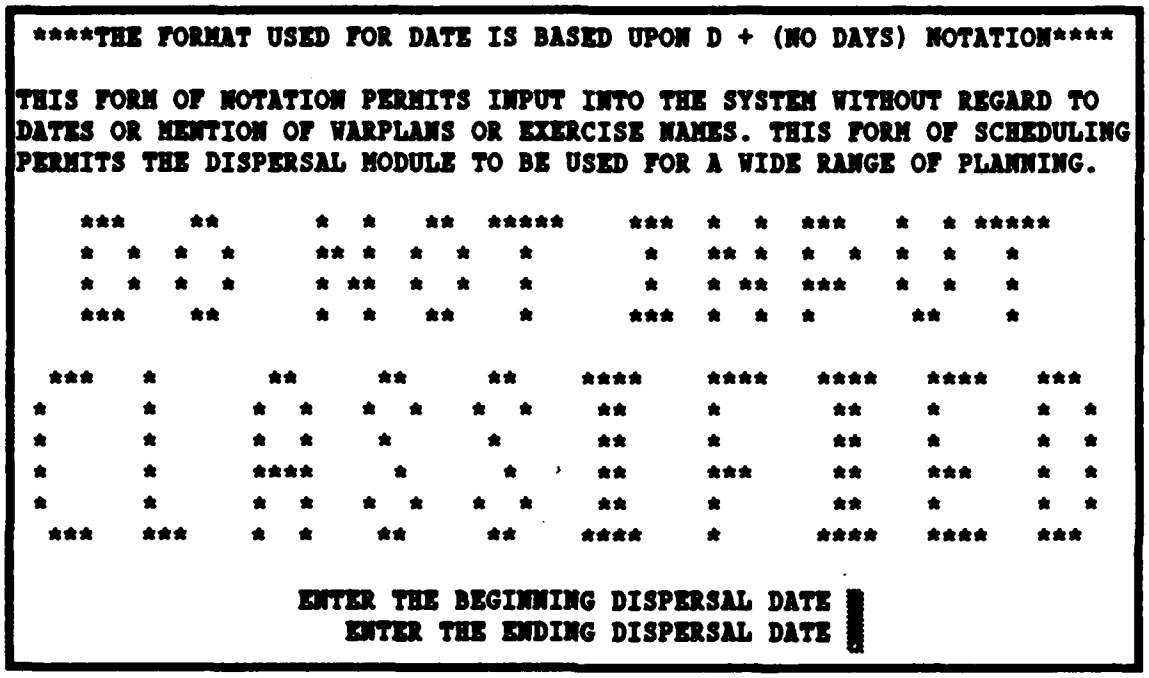


Fig. 22. Dispersal Classified Warning Screen

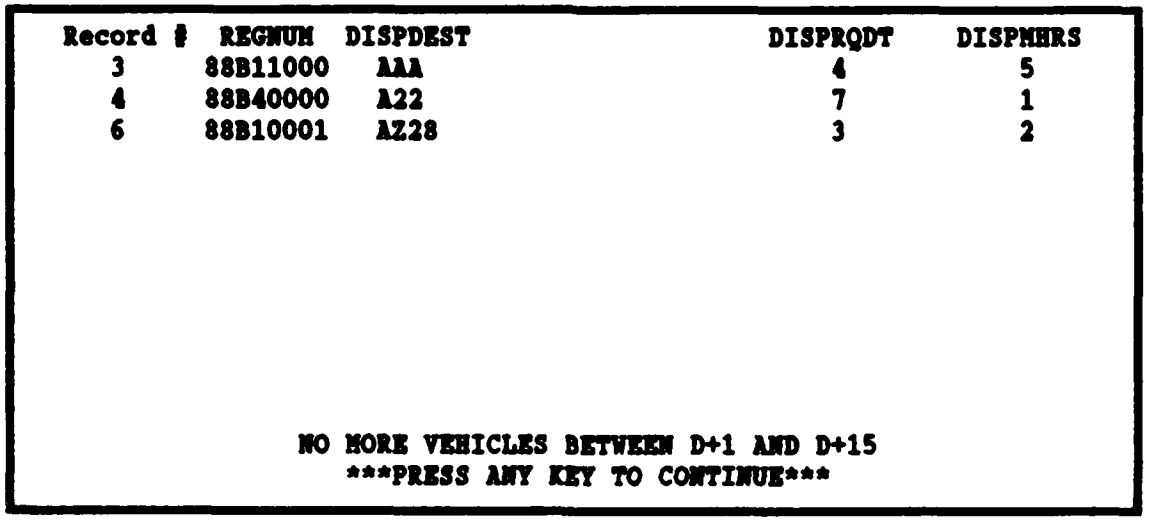


Fig. 23. Dispersal List by Date Display Screen

Scheduled Actions Management

The scheduled actions module uses the pkplan, disdis, exercise, release, manmaint, vims, and master databases to display all date dependent vehicle activities in a variety of different formats. The user enters the scheduled actions module by selecting option "3" from the System Main Menu as is portrayed in figure 24.

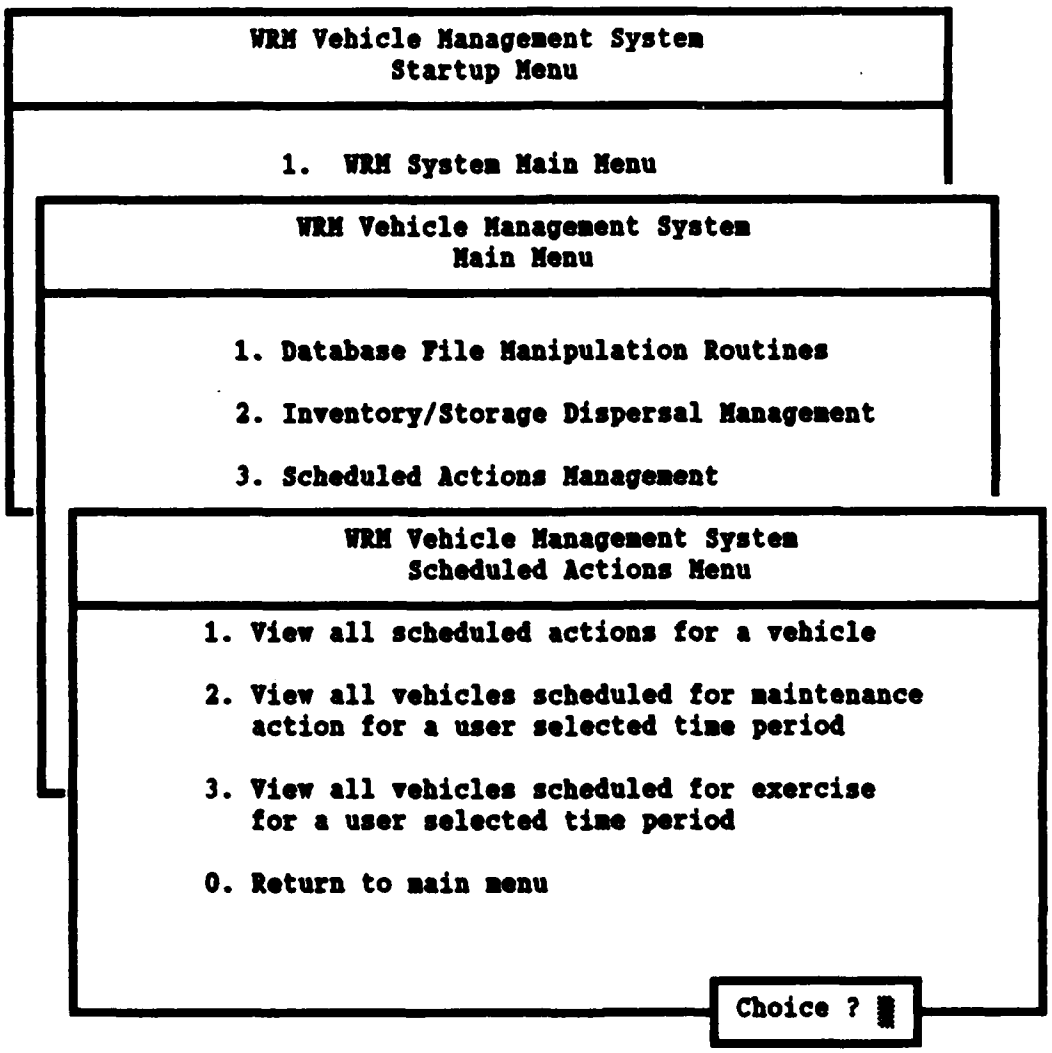


Fig. 24. Scheduled Actions Menu

If option "1" of the scheduled actions menu is selected, the user is requested to enter the vehicle registration number of the vehicle that is to be displayed. Option "1" does not offer the user the opportunity to produce printed output as the dispersal/distribution module does. The scheduling report and scheduling checklist options of the reports module will provide printed output containing the same information for the entire fleet, a subset of the fleet, or for an individual vehicle. Therefore, to avoid redundancy, this option was not offered in this module. Once the user has entered the vehicle registration number all scheduled actions for that vehicle are displayed to the screen as depicted in figure 25.

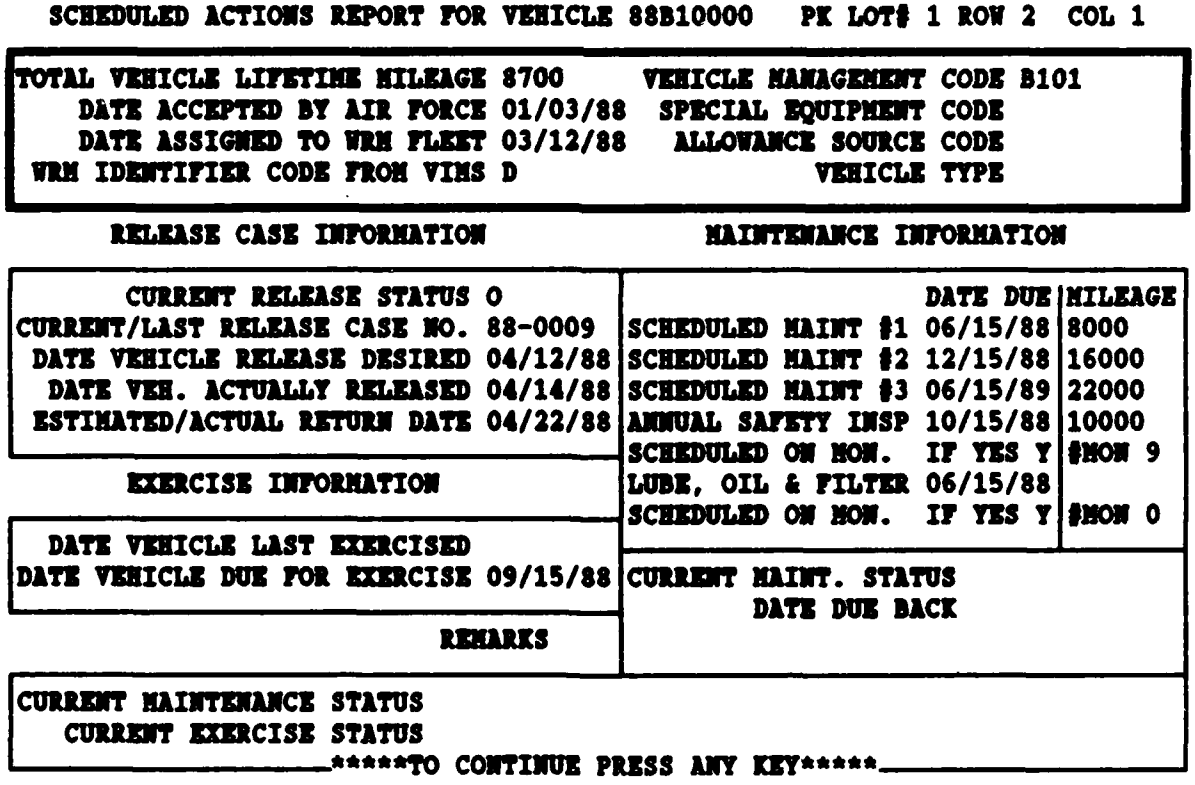


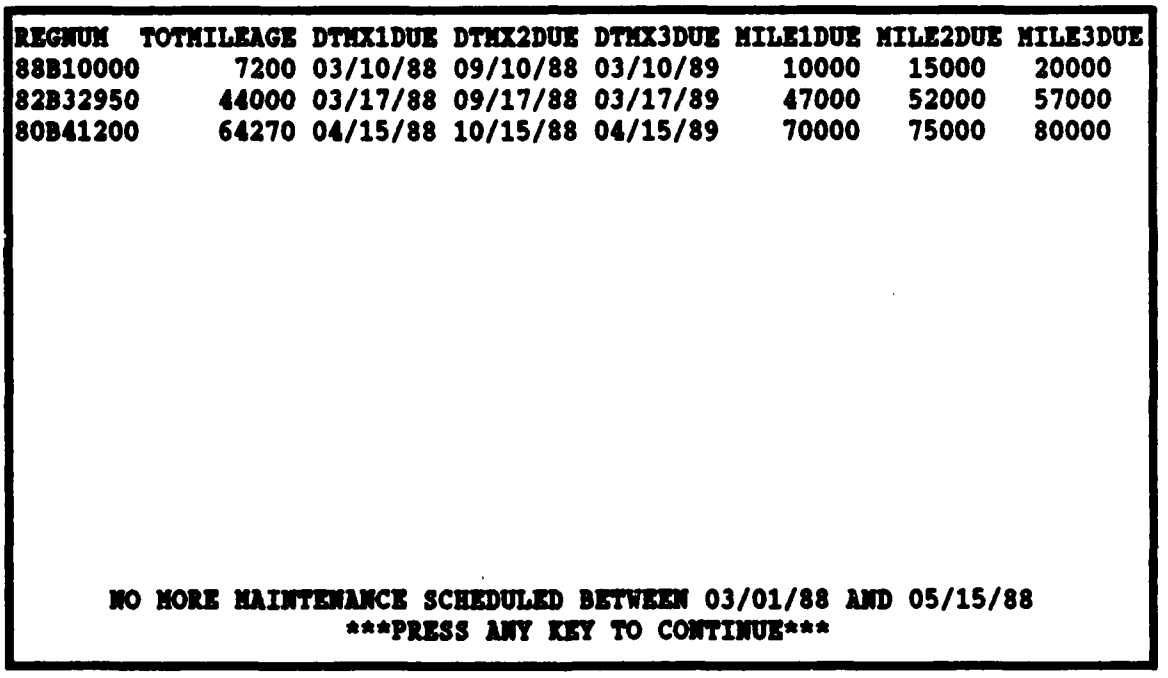
Fig. 25. Scheduled Actions View Screen

The primary purpose for presenting a screen that is so rich in information, is to permit the user to view at a glance any scheduling conflicts that are normally hidden in the sheer volume of data. Most of this information is usually kept in separate files or reports and requires transportation personnel to spend an inordinate amount of time cross-referencing pending scheduled actions. The net result is often a reduction in fleet readiness and some needless duplication of tasks. The scheduled actions module can search six different databases, and display all scheduled actions for a single vehicle in less than 10 seconds.

The primary source for scheduled maintenance type actions is the vims database. Options "2" and "3" of the scheduled actions module allows the user to sample the vehicle database for any subset of the fleet that is due scheduled maintenance or an exercise during a user selected time period.

Option "2" provides the user the option to view or print all scheduled maintenance actions for a selected time period. Once the beginning and ending dates are entered, the vims database is searched for any entries in the three scheduled maintenance due fields that fall in the selected time period. If the user selects a wide time range, more than one maintenance action may be due on the same vehicle. The effectiveness of this listing is therefore limited to a time period that is less than the routine maintenance cycle. For example, if the vehicle is scheduled for maintenance every six months, the selected time period should be less than six

months. The majority of the time the user will only be interested in the scheduled actions that fall in a 30 to 90 day range from the current date. A narrow range can be selected for a time period farther out in the future for planning purposes. The effective limit should be considered to be one year or less, because the Vehicle Integrated Management System does not project maintenance actions due beyond a year. Figure 26 depicts the maintenance display screen.



REGNUM	TOTMILEAGE	DTMX1DUE	DTMX2DUE	DTMX3DUE	MILE1DUE	MILE2DUE	MILE3DUE
88B10000	7200	03/10/88	09/10/88	03/10/89	10000	15000	20000
82B32950	44000	03/17/88	09/17/88	03/17/89	47000	52000	57000
80B41200	64270	04/15/88	10/15/88	04/15/89	70000	75000	80000

Fig. 26. Scheduled Maintenance View Screen

Option "3" permits the user to view all vehicles due for exercise during a selected time period. Vehicles in the WRM fleet must be exercised periodically to ensure their operability. This option will display to the screen or print a list of vehicles due for exercise in a user selected time

period. The option will display 20 vehicles to the screen and wait for the user to press any key to continue. Figure 27 depicts the option "3" display screen.

```
REGNUM    EXLSTDT  EXNXTDT                    EXRMKS
85B22000 03/20/88 07/20/88  VEHICLE USED FOR REFORGER
87B15670 03/20/88 07/20/88  VEHICLE USED FOR REFORGER
87B15671 04/10/88 08/10/88  VEHICLE USED IN EXERCISE 88-04
88B22459 04/10/88 08/10/88  VEHICLE USED IN EXERCISE 88-04
88B14010 05/20/88 09/20/88

NO MORE VEHICLES DUE FOR EXERCISE FROM 07/01/88 TO 10/01/88
***PRESS ANY KEY TO CONTINUE***
```

Fig. 27. Scheduled For Exercise View Screen

Release Case Management

The release case management module is designed to provide the capability to view and print all open release cases, a listing by release category , by using organization, and individual release case. Vehicles released from the WRM fleet for routine or emergency purposes are rigidly controlled and must be monitored daily. Vehicle release case trends may suggest the need for changes in the U-Drive it (UDI) fleet composition, maintenance scheduling problems, or approval of additional vehicles for organizations that continually need

vehicles released from the WRM fleet to satisfy operational needs. The user enters the release case management module by selecting option "4" of the WRM Vehicle Management System main menu as depicted in figure 28.

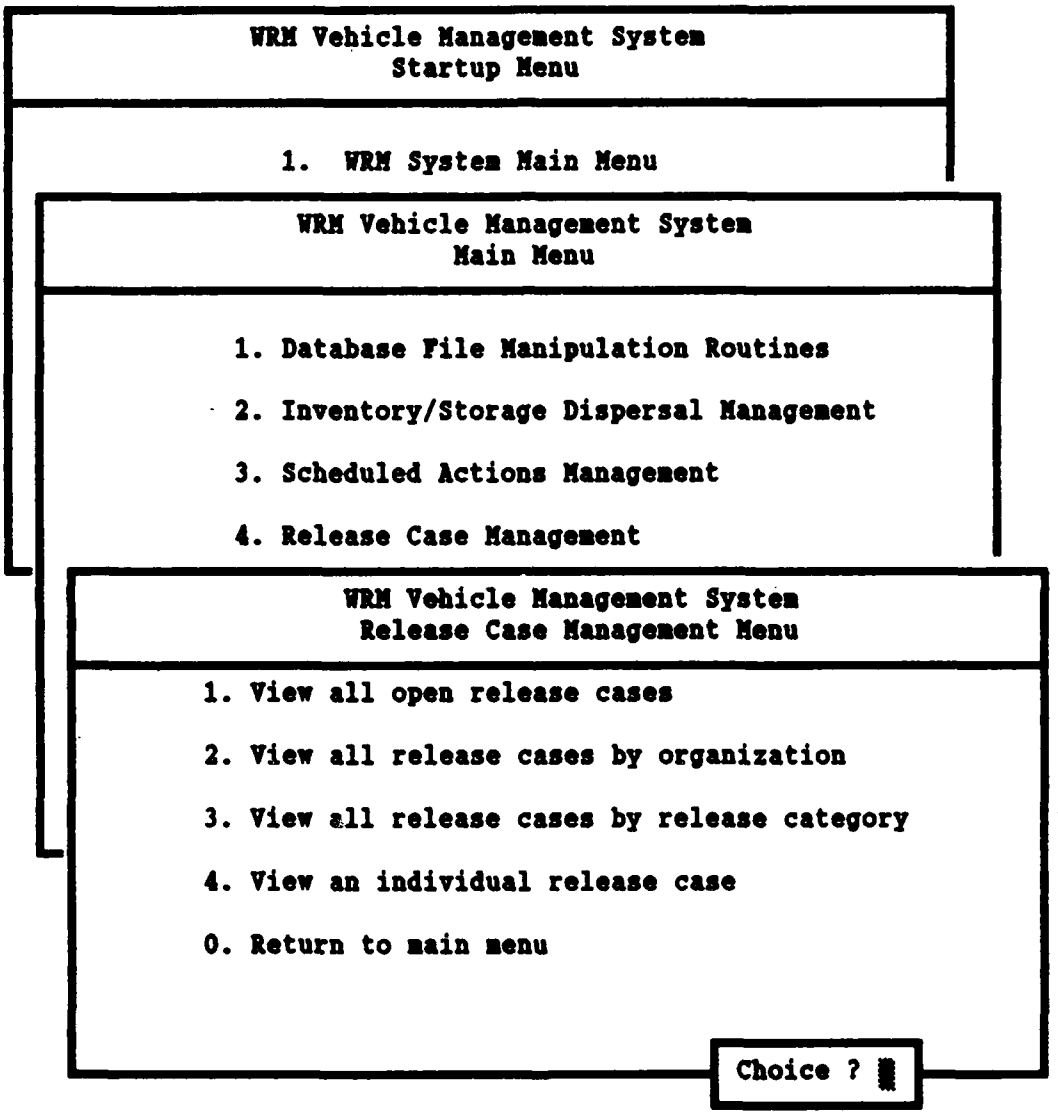
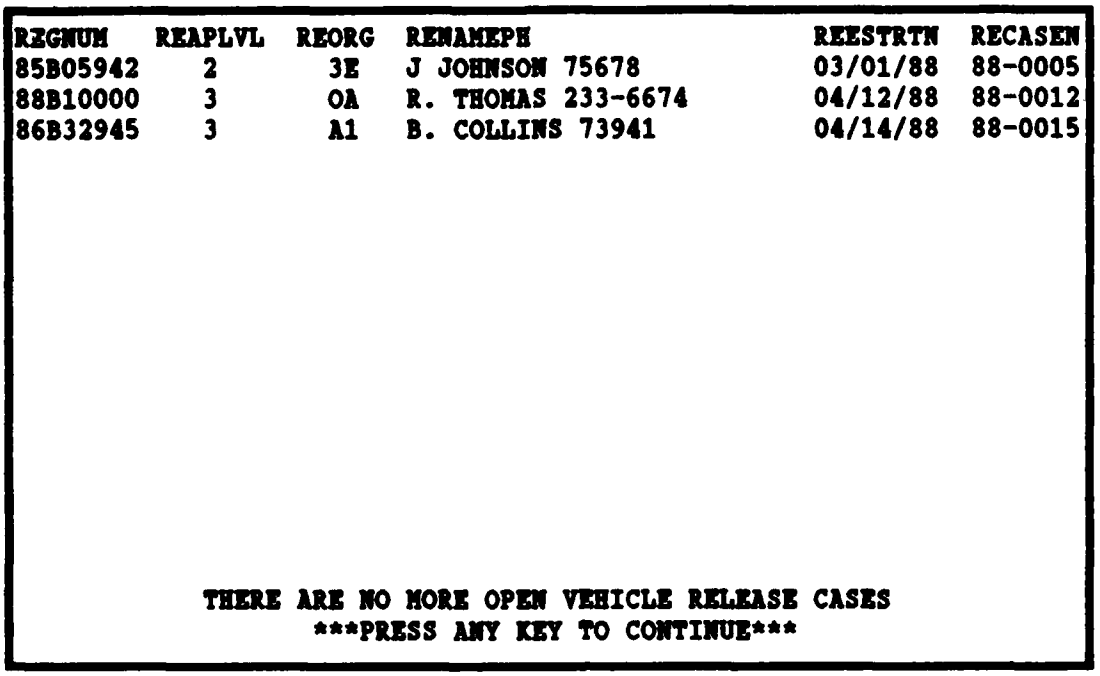


Fig. 28. Release Case Management Menu

Option "1" of the release case module permits the user to view all open release cases. Vehicle release cases are coded "O" for open or "C" for closed. The user is given the option

to view or print all open release cases. Figure 29 depicts the view screen for all open release cases.



REGNUM	REAPLVL	REORG	RENAMEPH	REESTRTN	RECASEN
85B05942	2	3E	J JOHNSON 75678	03/01/88	88-0005
88B10000	3	OA	R. THOMAS 233-6674	04/12/88	88-0012
86B32945	3	A1	B. COLLINS 73941	04/14/88	88-0015

THERE ARE NO MORE OPEN VEHICLE RELEASE CASES
PRESS ANY KEY TO CONTINUE

Fig. 29. Open Release Case View Screen

The open release case view screen does not present all the information available in the release database. The current release case status report in the reports module presents an in-depth review of all open release cases. The view screen displayed in figure 29 is designed to present information pertinent to the daily WRM fleet management activities while the current release case status report is more appropriate for contingency exercises or dispersals.

Option "2" of the release case management module provides the user the ability to view or print all open release cases by organization. This option is quite helpful during contingency exercises when rapid up-to-date information about how many vehicles has been released to specific organizations

can increase the efficiency of allocating resources. This option can also aid transportation personnel in identifying organizations that need increased vehicle authorizations to meet their operational requirements. Figure 30 depicts the release case by organization view screen.

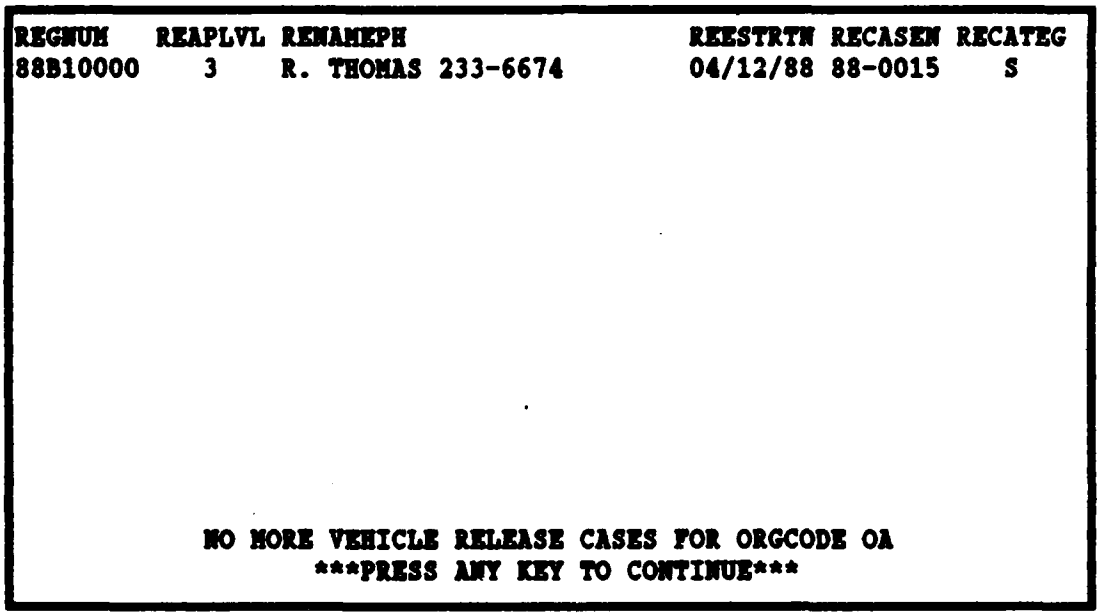


Fig. 30. Release Case by Organization View Screen

The selection of option "3" permits the user to view and print all open and closed release cases by release category. Any alphanumeric release code can be entered, however, the summary report provided in the reports module recognizes only the following release category codes:

1. "D" - release for a disaster
2. "E" - release for an emergency
3. "X" - release for an exercise
4. "S" - release to meet a surge requirement
5. "R" - release in the rental category

6. "M" - release in a miscellaneous category

If alternate codes are used, option "3" is the most convenient method to total the number of open and closed release cases. When the output of option "3" is combined with the reports module release case analysis report, release case analysis is simplified. The release case by release category view screen is presented in figure 31.

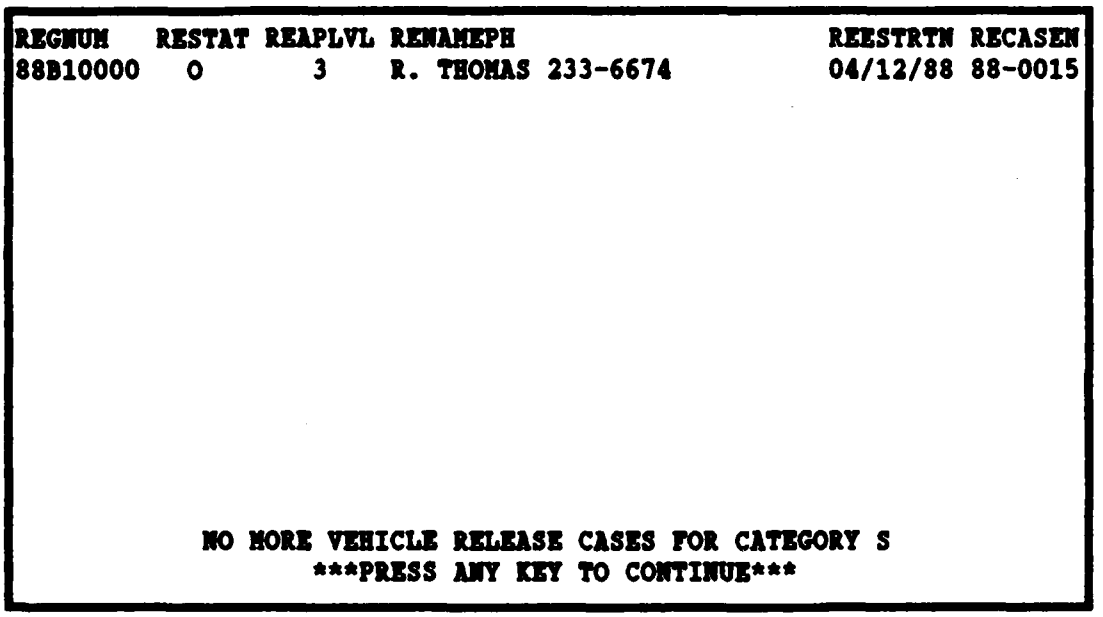


Fig. 31. Release Case by Release Category View Screen

Option "4" of the release case management module permits the user to view all release case information for a vehicle. The user is requested to enter the vehicle registration number to view all fields of information in the release database. This same option is presented in the database routines module to permit the user to view recently entered data. The view option is presented in this module to permit the user to view some fields of information that are not presented in any of

the other options in this module. The option to print this information is not offered because the same information is available in the release case status report of the reports module. The release case view screen is presented in figure 32.

```
                WRM VEHICLE MANAGEMENT SYSTEM
            VIEW A VEHICLE RECORD (RELEASE DATABASE)

          VEHICLE REGISTRATION NUMBER 88B10000
               CURRENT RELEASE STATUS O          [O]pen,[C]losed
               CURRENT APPROVAL LEVEL 3
ORGANIZATION CODE VEHICLE RELEASED TO OA
NAME AND PHONE NUMBER OF VEHICLE USER T. Howard 73957
          DATE VEHICLE RELEASE DESIRED 04/10/88  (MM/DD/YY)
        DATE VEHICLE ACTUALLY RELEASED 04/10/88  (MM/DD/YY)
          ESTIMATED/ACTUAL RETURN DATE 04/14/88  (MM/DD/YY)
         CURRENT/LAST RELEASE CASE NO. 88-0052   Ex. 88-1000
    CURRENT/LAST RELEASE CASE CATEGORY E

                          REMARKS
          VEHICLE RELEASED TO SATISFY EMERGENCY REQUIREMENT

                  ***PRESS ANY KEY TO CONTINUE***
```

Fig. 32. Vehicle Release Case View Screen

Reports module

The reports module of the WRM Vehicle Management System provides a variety of printed reports. Some reports are available for individual vehicles while other reports are only available for the entire fleet. The only reports that permit the user to print the data for a single vehicle are designed for use as scheduling checklists. The database routines module permits the user to view vehicle data for every database. Through the use of the shift/print screen option, the user can produce printed output for any single vehicle by

entering the view option then pressing the shift and print screen keys simultaneously. Thus, the reports module is primarily oriented towards reports for the whole fleet. The user enters the reports module by entering choice "5" on the system main menu as depicted in figure 33.

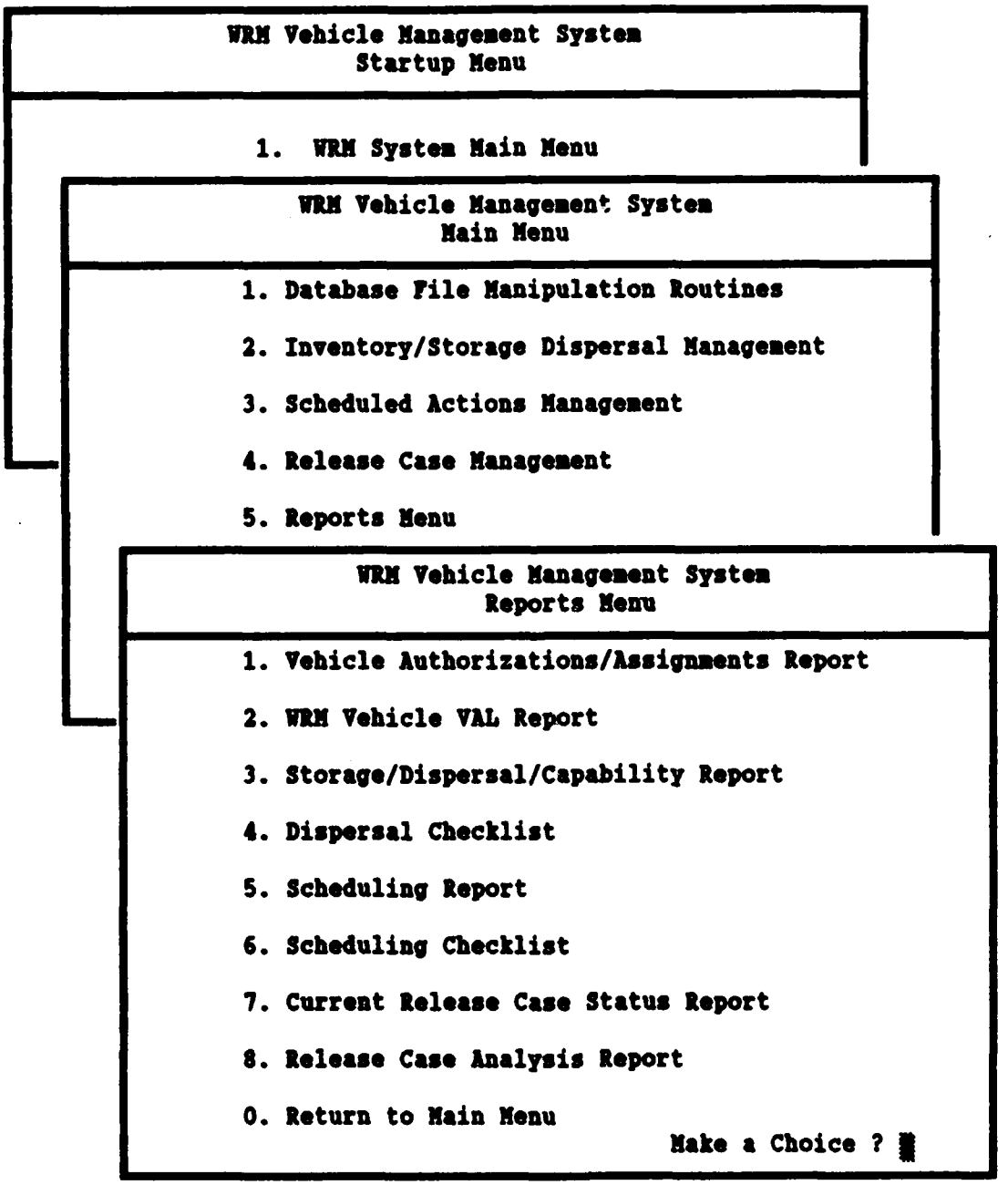


Fig. 33. Reports Menu View Screen

The vehicle authorizations/assignments report, initiated by selecting option "1" on the reports menu, uses the master and summary database to satisfy the requirement for up-to-date information on fleet status. This report is offered for the entire fleet. Figure 34 portrays a sample page of the vehicle authorizations assignments report.

Vehicle Authorizations/Assignments
Report as of 05/10/88

VEHICLE:78B01645

MGTCODE B200	ITEM CODE S
ASGN MGTCODE B200	REP CODE G
MASTER NSN 2320-01-124-7517	USE CODE M
VEHICLE NSN 2320-01-009-6194	DATE ASGN 03/10/78
ASGN NSN 2320-00-999-9999	MOBL CODE J
NOMENCLATURE TRK PU CMPT 4X2	ASC 010A00C
ORGCODE 4N	NO EXS TYEAR 3
OWN CMD 0S	NO REL TYEAR 6
USE CMD 0S	REL DAYS TOT 41

VEHICLE:77E00158

MGTCODE E832	ITEM CODE S
ASGN MGTCODE E820	REP CODE N
MASTER NSN 3930-010032-3026	USE CODE M
VEHICLE NSN 2320-01-009-6194	DATE ASGN 07/11/77
ASGN NSN 3930-00-879-2157	MOBL CODE J
NOMENCLATURE FKLFT ACFT LOADR	ASC 010A00D
ORGCODE 20	NO EXS TYEAR 1
OWN CMD 0H	NO REL TYEAR 0
USE CMD 0H	REL DAYS TOT 0

VEHICLE:78B01647

MGTCODE B200	ITEM CODE S
ASGN MGTCODE B200	REP CODE G
MASTER NSN 2320-01-124-7517	USE CODE M
VEHICLE NSN 2320-01-009-6194	DATE ASGN 03/10/78
ASGN NSN 2320-00-999-9999	MOBL CODE J
NOMENCLATURE TRK PU CMPT 4X2 45	ASC 010A00C
ORGCODE 6A	NO EXS TYEAR 2
OWN CMD 0H	NO REL TYEAR 3
USE CMD 0H	REL DAYS TOT 8

PAGE 1

Fig. 34. Vehicle Authorizations/Assignments Report

Option "2" of the reports module offers the ability to print a vehicle authorizations listing (VAL) for all WRM vehicles. The WRM VAL is used to determine WRM vehicle fill rates, identify shortfalls, and assist in monitoring fleet status. Figure 35 portrays a sample page of the WRM vehicle authorization listing.

```
                WRM Vehicle Authorization Listing
                     Report as of 05/10/88

                  VEHICLE:87B05550

VAL STOCK NO 2320-01-009-6194          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                      NUMBER AUTH 6
VAL ORG CODE 20                           NUMBER ASGN 6
  PRI RECALL 1
-------------------------------------------------------------

                  VEHICLE:78B01645

VAL STOCK NO 2320-01-124-7517          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                      NUMBER AUTH 6
VAL ORG CODE 20                           NUMBER ASGN 6
  PRI RECALL 2
-------------------------------------------------------------

                  VEHICLE:76B02387

VAL STOCK NO 2320-01-009-6194          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                      NUMBER AUTH 6
VAL ORG CODE 20                           NUMBER ASGN 6
  PRI RECALL 1
-------------------------------------------------------------

                  VEHICLE:78B01647

VAL STOCK NO 2320-01-124-7517          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                      NUMBER AUTH 6
VAL ORG CODE 20                           NUMBER ASGN 6
  PRI RECALL 1
-------------------------------------------------------------

                        PAGE 1
```

Fig. 35. WRM Vehicle Authorization Listing Report

Option "3" of the Reports module, the storage/dispersal/ capability report, provides the user the ability to plan vehicle distribution and dispersal. Figure 36 displays a sample of the storage/dispersal/capability report.

```
                    Storage/Dispersal/Capability
                       Report as of 05/10/88

          VEHICLE:78B01645

 PK LOT NO: 2      PK LOT ROW: 2     PK LOT COL: 4     PARKING STATUS: S

      MGTCODE B200                        REP CODE G
  VEHICLE NSN 2230-01-124-7517            USE CODE M
NOMENCLATURE TRK PU CMPT 4X2            DATE ASGN 03/10/78
     ORG CODE 4N                        MOBL CODE J
      OWN CMD 0S                              ASC 010A00C
      USE CMD 0S                    TOT MILEAGE 52000
    ITEM CODE S                         VEH TYPE M

                        DISPERSAL INFORMATION

 DISP DEST AZ322                            DISP RQD DATE: 6
 DISP MHRS 3
 DISP INST:
 DISP RMKS:
-------------------------------------------------------------------------

          VEHICLE:77E00158

PK LOT NO: 1      PK LOT ROW: 4     PK LOT COL: 4     PARKING STATUS: P

      MGTCODE E832                        REP CODE S
  VEHICLE NSN 3930-010032-3026            USE CODE M
NOMENCLATURE FKLFT ACFT LOADR           DATE ASGN M
     ORG CODE 20                        MOBL CODE J
      OWN CMD 0H                              ASC 010A00D
      USE CMD 0H                    TOT MILEAGE
    ITEM CODE S                         VEH TYPE H

                        DISPERSAL INFORMATION

 DISP DEST AN23                             DISP RQD DATE: 1
 DISP MHRS 5
 DISP INST: MUST DISPERSE WITH 463L TINES
 DISP RMKS:
-------------------------------------------------------------------------

                              PAGE 1
```

Fig. 36 Storage/Dispersal/Capability Report

The dispersal checklist can be generated by selecting option "4" of the reports module. This report is provided in checklist form to aid rapid dispersal. The dispersal checklist is provided for a vehicle and not the whole fleet. The dispersal checklist is depicted in figure 37.

```
                    DISPERSAL CHECKLIST
                      AS OF 05/10/88

          VEHICLE REGISTRATION NUMBER  78B01645
              VEHICLE MANAGEMENT CODE  B200
                   ORGANIZATION CODE   6A
                OWNING MAJOR COMMAND   0H
                 USING MAJOR COMMAND   0H
                WRM VEHICLE USE CODE   M
             WRM IDENTIFIER FROM VIMS   J
               ALLOWANCE SOURCE CODE   010A00C
        VEHICLE MILE/KM/HR/UNIT TYPE   M
                  PARKING LOT NUMBER   3
                  PARKING ROW NUMBER   5
               PARKING COLUMN NUMBER   2
                     PARKING STATUS    P
                DISPERSAL DESTINATION  AZ1522
DATE REQUIRED AT WAR TIME LOCATION     21          (D+DAY FORMAT)
     MANHOURS NEEDED TO DISTRIBUTE     5

                      DISPERSAL REMARKS

                  SPECIAL DISPERSAL INSTRUCTIONS
          VEHICLE MUST BE EQUIPPED WITH PENTLE HOOK WHEN DISPERSED
```

Fig. 37. Dispersal Checklist

The scheduling report, option "5", contains all date dependent scheduling activities for vehicles in the fleet. This report produces the same information that option "1" of the scheduled actions menu displays to the screen. The user may view all scheduled actions for a single vehicle. The scheduling report is printed for all vehicles in the fleet. The scheduled actions report is depicted in figure 38.

```
SCHEDULED ACTIONS REPORT FOR VEHICLE 88B10000   PK LOT# 1 ROW 2  COL 1

 TOTAL VEHICLE LIFETIME MILEAGE 8700      VEHICLE MANAGEMENT CODE B101
     DATE ACCEPTED BY AIR FORCE 01/03/88   SPECIAL EQUIPMENT CODE
     DATE ASSIGNED TO WRM FLEET 03/12/88    ALLOWANCE SOURCE CODE
  WRM IDENTIFIER CODE FROM VIMS D                    VEHICLE TYPE

       RELEASE CASE INFORMATION              MAINTENANCE INFORMATION

        CURRENT RELEASE STATUS O                              DATE DUE MILEAGE
 CURRENT/LAST RELEASE CASE NO. 88-0009  SCHEDULED MAINT #1 06/15/88 8000
  DATE VEHICLE RELEASE DESIRED 04/12/88 SCHEDULED MAINT #2 12/15/88 16000
   DATE VEH. ACTUALLY RELEASED 04/14/88 SCHEDULED MAINT #3 06/15/89 22000
  ESTIMATED/ACTUAL RETURN DATE 04/22/88 ANNUAL SAFETY INSP 10/15/88 10000
                                        SCHEDULED ON MON.  IF YES Y #MON 9
        EXERCISE INFORMATION            LUBE, OIL & FILTER 06/15/88
                                        SCHEDULED ON MON.  IF YES Y #MON 0
   DATE VEHICLE LAST EXERCISED
 DATE VEHICLE DUE FOR EXERCISE 09/15/88 CURRENT MAINT. STATUS
                                              DATE DUE BACK
                            REMARKS

 CURRENT MAINTENANCE STATUS
    CURRENT EXERCISE STATUS
```

```
  SCHEDULED ACTIONS REPORT FOR VEHICLE 85B30569   PK LOT# 1 ROW 2  COL 2

 TOTAL VEHICLE LIFETIME MILEAGE 19600     VEHICLE MANAGEMENT CODE B200
     DATE ACCEPTED BY AIR FORCE 01/03/88   SPECIAL EQUIPMENT CODE
     DATE ASSIGNED TO WRM FLEET 03/12/88    ALLOWANCE SOURCE CODE 010A00C
  WRM IDENTIFIER CODE FROM VIMS J                    VEHICLE TYPE M

       RELEASE CASE INFORMATION              MAINTENANCE INFORMATION

        CURRENT RELEASE STATUS C                              DATE DUE MILEAGE
 CURRENT/LAST RELEASE CASE NO. 87-0018  SCHEDULED MAINT #1 06/15/87 18000
  DATE VEHICLE RELEASE DESIRED          SCHEDULED MAINT #2 12/15/87 24000
   DATE VEH. ACTUALLY RELEASED 03/01/87 SCHEDULED MAINT #3 06/15/88 30000
  ESTIMATED/ACTUAL RETURN DATE 04/11/87 ANNUAL SAFETY INSP 10/15/87 15000
                                        SCHEDULED ON MON.  IF YES N  #MON
        EXERCISE INFORMATION            LUBE, OIL & FILTER           18000
                                        SCHEDULED ON MON.  IF YES N #MON 0
   DATE VEHICLE LAST EXERCISED 03/10/87
 DATE VEHICLE DUE FOR EXERCISE 03/10/88 CURRENT MAINT. STATUS
                                              DATE DUE BACK
                            REMARKS

 CURRENT MAINTENANCE STATUS
    CURRENT EXERCISE STATUS
```

PAGE 1

Fig. 38. Scheduling Report

The scheduling checklist is selected by entering option "6" in the reports module. The scheduling checklist is designed for use as a worksheet when vehicles are being inspected, exercised, or maintained. The user is prompted for a vehicle registration number, which when entered will generate the checklist presented in figure 39.

Scheduling Checklist
as of 05/10/88

VEHICLE:85B30569

VEHICLE MANAGEMENT CODE: B200
SPECIAL EQUIPMENT CODE: AZZ09876ZZ
VEHICLE MILE/KM/HR/UNIT TYPE: M
DATE SCHEDULED MAINTENANCE #1: 18000
DATE SCHEDULED MAINTENANCE #2: 24000
DATE SCHEDULED MAINTENANCE #3: 30000
DATE ANNUAL INSPECTION SCHED: 08/15/88
DATE LUBE, OIL & FILTER DUE: 08/15/88
TOTAL VEHICLE LIFETIME MILEAGE: 21000
MILEAGE MAINTENANCE #1 DUE: 18000
MILEAGE MAINTENANCE #2 DUE: 24000
MILEAGE MAINTENANCE #3 DUE: 30000
CURRENT MAINTENANCE STATUS: C
DATE VEHICLE DUE BACK FROM MAINT:
CURRENT RELEASE CASE STATUS: C
ESTIMATED/ACTUAL RETURN DATE:
DATE VEHICLE LAST EXERCISED: 03/10/87
DATE VEHICLE DUE FOR EXERCISE: 03/10/88
PARKING LOT NUMBER: 1
PARKING ROW NUMBER: 2
PARKING COLUMN NUMBER: 2
PARKING STATUS: P

Fig. 39. Scheduling Checklist

Option "7" provides the user with the option to print the current release case status report for all open release cases. This report is designed to meet the requirement for up-to-date release case information during contingency exercises. Figure 40 depicts the current release case status report.

CURRENT RELEASE CASE STATUS
REPORT AS OF 05/10/88

VEHICLE IDENTIFICATION INFORMATION

REGNUM	MGT CODE	NOMENCLATURE	ORG CODE	OWN CMD	USE CMD	SPEQCODE	MOBL CODE	ASC
85B05942	B204	PASSENGER VAN 9 PAX	4M	OS	OS	AZZ09876ZZ	J	010A00C

RELEASE CASE INFORMATION

CASE NUMBER	USER NAME, PHONE NUMBER	USER ORG	RQD DATE	ACCPT DATE	EST RTN
88-0050	R JOHNSON 75678	3E	04/11/88	04/11/88	05/27/88

REMARKS:

PARKING INFORMATION

LOT NUMBER	ROW NUMBER	COLUMN NUMBER	PARK STATUS
1	5	3	P

VEHICLE IDENTIFICATION INFORMATION

REGNUM	MGT CODE	NOMENCLATURE	ORG CODE	OWN CMD	USE CMD	SPEQCODE	MOBL CODE	ASC
85b30569	B200	TRK PU CMPT 4X2	4N	OS	OS	AZZ09876ZZ	J	010A00C

RELEASE CASE INFORMATION

CASE NUMBER	USER NAME, PHONE NUMBER	USER ORG	RQD DATE	ACCPT DATE	EST RTN
88-0052	B. LANDRY 74389	3E	05/18/88	06/20/88	06/20/88

REMARKS:

PARKING INFORMATION

LOT NUMBER	ROW NUMBER	COLUMN NUMBER	PARK STATUS
2	2	4	T

PAGE 1

Fig. 40. Current Release Case Report

The final option in the reports module is option "8", the release case analysis report. The release case analysis report provides a summary of historical release case data. The purpose for this report is to help the user analyze trends that may indicate a need for reallocating vehicle assets. Figure 41 depicts the release case analysis report. The report is presented in condensed print as in the actual report.

Release Case Analysis Report
as of 06/10/88

REGNUM	MGT CODE	ORG CODE	OWN CND	USE CND	SPEQ CODE	ASC	REL APV LVL	REL THIS YEAR	REL TOT. DAYS	DAYS "D" CAT1	DAYS "E" CAT2	DAYS "I" CAT3	DAYS "S" CAT4	DAYS "R" CAT5	DAYS "M" CAT6	NUMB "D" CAT1	NUMB "E" CAT2	NUMB "I" CAT3	NUMB "S" CAT4	NUMB "R" CAT5	NUMB "M" CAT6
88B10000	B101	6A	OK	OK		010A00C	3	4	11		3	8					2	2			
78B01645	B200	4M	OS	OS		010A00C	2	1	4			4						1			
77E00156	E832	20	OK	OK		010A00B	3	2	5		1	4					1	1			
						TOTALS	8	7	20		4	16					3	4			

PAGE 1

Fig. 41. Release Case Analysis Report

System Utilities

The system utilities module provides system level functions that enable the user to automatically input and update the WRM system databases, pack and index the databases, and copy all database files. The user enters the systems utilities menu by selecting option "2" on the startup menu as depicted in figure 42.

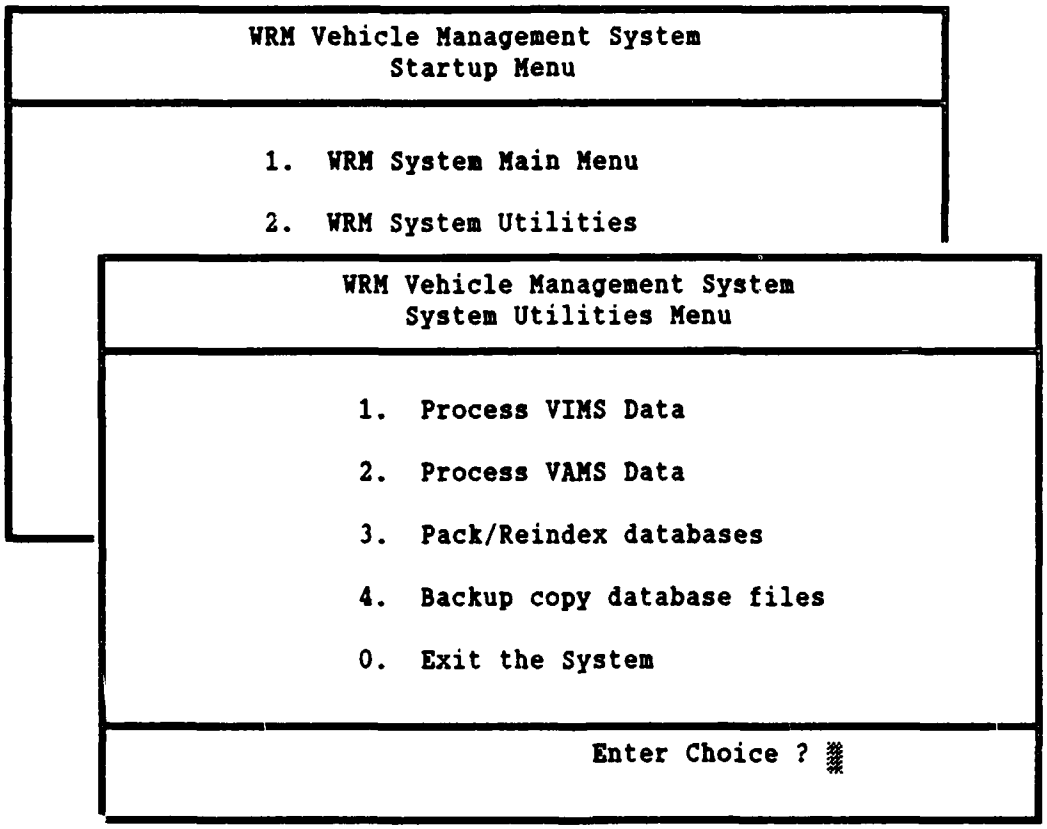


Fig. 42. System Utilities Menu

The user may process data from the Vehicle Integrated Management System (VIMS) by entering choice "1" on the systems utilities menu. The VIMS data files must be down-loaded from the Sperry 1100 mainframe, or received from the On Line VIMS (OLVIMS) Zenith Z248™ microcomputer. The down-loading

procedure from the Sperry 1100 system is somewhat complicated, and step by step procedures are provided in the user manual located in appendix "C" of this thesis. The down-loading procedure for OLVIMS data is automatically accomplished during the OLVIMS processing, and is in a format acceptable to the VAMS system. The OVIMS data is converted into the WRM system format under program control during OLVIMS processing.

When option "1" is selected, the user is requested to select the source the data is being transferred from, VIMS or OLVIMS. Upon entering the source, the user is instructed to place the floppy disk containing the VIMS data in disk drive "A", and press return. The screen will clear, and a message will appear stating that VIMS processing has begun. The data is being converted to the appropriate format suitable for use by the WRM vehicle management system. The VIMS data-file can now be copied to the "C" drive if a hard disk drive is being used, or to the appropriate data disks. This process should occur monthly, and the user should retain the previous months data files, to reconcile errors by comparing output from both database files.

Option "2" of the system utilities menu permits the user to convert Vehicle Automated Management System files into WRM system files. As mentioned in chapter three, this option is superior to accepting data directly from the VIMS system, because reconciliation of the different databases has already occurred. Because the data has already been "scrubbed" for errors, the WRM system user need only select option "2" and

place the VAMS floppy disk in disk drive "A" for the file conversion to take place. Upon selecting choice "2", the user is requested to place the VAMS data disk in disk drive "A", and press return. The screen will clear, and a message will appear advising that VAMS conversion has begin. Upon completion, a message will appear instructing the user that the file conversion is now complete and ready for WRM system use.

The advantage associated with automatic update of the VAL database are not so significant that this system can not be effectively used without these procedures. Once the user has manually entered all the vehicle data through the database routines module, very little changes in the VAL data actually occurs. The user can manually enter changes through the edit options with little effort. Printed output is available through the WRM vehicle VAL report in the reports module. If the WRM fleet is small enough, it may be advantageous to build and update the databases manually.

Option "3" of the system utilities menu, offers the user the ability to pack and reindex the databases. This option will permanently remove vehicles marked for deletion in all of the databases. If a database index file has inadvertently been corrupted by interrupting a database operation (power outage), the pack/reindex option can rebuild all index files needed by the system to operate. Thus, the reindex portion of the pack operation is a very important tool for the WRM system user to become familiar with. If the WRM system indicates an

error in reading a database file during normal operation, the error will be in all probability with the index file of that database. The user manual instructs the user to reindex the files whenever database error messages occur.

Once the user selects option "3", the screen clears and a message appears to insert the data disk in disk drive "A" and press return. If the WRM system has been installed on a hard disk, the user need only press return and all data-files will be packed and reindexed. If the WRM system is being operated on a floppy disk system and more than one data disk exists, the user must repeat this operation for each data disk and each data disk must contain all eight databases with the appropriate vehicle information. This does not present a problem, as the user manual instructs the user to copy the data disk as many times as is necessary to hold all vehicle data files. During the copy process all database and index files will be copied automatically to each data disk. The user is also instructed to never use the original data disks or program disks prior to making several backup copies.

Option "5" permits the user to prepare backup copies of the databases. If the user operates the WRM system using a floppy disk system, this option will not prove as beneficial as using the "DISKCOPY" option provided with the MS DOS™ system disks. If the user operates the WRM system using a hard disk system, this option will prove to be invaluable. The data-files created by the WRM system during normal operation with a hard disk, are only limited in size by the

size of the hard disk. This allows the database files to grow in size beyond the 360 kilobyte floppy disk storage limit. Thus, the hard disk user must apportion the database into smaller size files for backup to floppy disks. This can be done using MS DOS™ commands, however option "5" accomplishes this action without the user having do more than insert blank disks as they become full. The user is encouraged to back up the database files weekly. The adage that there are only two types of computer users, the ones who have lost data files or programs, and those who are going to lose data files or programs, makes this advice very valuable. When option "5" is selected, the user is advised to place a blank formatted floppy disk in disk drive "A" and press return. Upon pressing return, the screen clears, and a message appears advising that the data files are now being copied. As each data disk becomes full, a message appears advising the user to insert another blank floppy disk, or if no more files need copying that the copying process is complete.

V. Conclusions and Recommendations

Conclusions

This research began by investigating the possibility of developing a microcomputer database system to satisfy the documented deficiencies identified in the 1984-86 Air Force IG reports.

Five specific research questions were identified in chapter one that needed to be answered in order to develop a computer based MIS for WRM vehicle management.

The literature review revealed that the problems that existed with management of the WRM vehicle fleet were primarily associated with data management. A review of the Air Force Logistics Management Center's (AFLMC) functional description for a WRM vehicle management system, revealed that system requirements were thoroughly defined. Subsequent telephone contacts with USAFE and PACAF transporters confirmed that the AFLMC functional description needed no amendment, and that WRM vehicle management problems continued to exist.

The bulk of the literature review was spent in determining what type of programming language was appropriate for development of this system. The dBase III PLUS™ software package was selected because of its excellent programming features, and the availability of commercial compilers.

The programming effort consumed in excess of 2,000 hours of time to complete. The initial effort produced a program that consisted of 11,000 lines of code and a file that was over 300 kilobytes in size. The size of the program, and the

slow database access time, forced the program to be rewritten using common procedure procedure files.

The reprogramming effort resulted in a program that consumed 110 kilobytes of disk space, and consisted of 4,000 lines of code. The database access times were significantly enhanced and overall program execution improved. A compiler was then purchased, and a third rewrite was accomplished to match the idiosyncrasies of the compiler.

The resulting program was able to fit on a 5 1/4 inch floppy disk and remain RAM resident without making repeated function calls to the program disk. This capability opened the possibility for the program to operate on hard disk or floppy disk systems. The program was designed to run on the Air Force procured Zenith™ microcomputer systems. All printed output was written to function on any printer that can emulate the Epson™ standard. This eliminated the requirement to write printer drivers for a variety of different printer types.

The concept of "user friendliness" was implemented throughout the programming effort. The systems approach was adopted to develop system structure, and implemented through "top down design." The final product can be operated by enlisted transportation personnel with little microcomputer experience. A modular approach using menus to offer options reduces the amount of time necessary to learn the system.

Due to the amount of time required to develop the system, validation was limited to pre-field testing. Contacts were made with AFLMC representatives and HQ PACAF/LGT

transportation personnel to initiate field testing and implementation.

Recommendations

The resulting system can be used to manage WRM vehicles during peacetime and wartime. The ability to run the system on a portable floppy based system, invites the possibility to use the system at staging locations with follow on capability in the field under battle conditions. Vehicles could be moved to forward locations accompanied by the pertinent vehicle information contained on floppy disks, uploaded into the WRM vehicle system and subsequent dispersal could be effectively managed.

A thesis research effort coinciding with a major exercise like Reforger, could test the feasibility of planning to use this approach during battlefield conditions.

Another possible thesis effort could focus on rewriting an updated version of the VAMS system in dBase III PLUS™ to take advantage of the database language enhancements that have occurred since the VAMS system was written.

This research effort demonstrates the amount of potential that exists for end-user development of systems that are focused to handle specific problems. The researcher possessed no unique computer skills and had no experience with database programming languages. Management effectiveness in the transportation arena hinges on sound asset management techniques. The volume of data that must be manipulated to accomplish this end simply overwhelms manual techniques. The

opportunities exist in abundance for microcomputer based management systems that can meet this demand.

Other Applications The use of this system is not limited to WRM vehicle management. The peacetime fleet can also be managed using this system without any reprogramming. Some of the reports offered in the reports module could be very effective in management of the peacetime fleet.

The same program could easily be adapted to manage the Army's vehicle fleets with some reprogramming. The key field, the vehicle registration number, could be changed very easily to match the vehicle control function that the Army uses.

Finally, this program could be altered to provide an "expert choice" option that would enhance the senior transportation manager's ability to perform realistic "what if" analysis.

Appendix A: Program Coding

```
*********************************
*         STARTUP.PRG           *
* startup module for WRM Prgm *
*********************************

*****************************************
PROGRAMMER: ROBERT THOMAS 1988
     CALLS: PROCA.PRG, MAIN.PRG, UTIL.PRG
*****************************************
**ENVIRONMENT**
set talk off
set escape off
************************************************
***LOADING MESSAGE***
● 0,0 CLEAR
● 11,21 TO 13,57 DOUBLE
● 12,23 SAY "LOADING PROGRAM PLEASE BE PATIENT"
************************************************
***DECLARE VARIABLES PUBLIC FOR USE BY ALL MODULES***
*************************************************************************
PUBLIC MREGNUM,MTOTMILAGE,MVEHTYPE,MDTMX1DUE,MDTMX2DUE,MDTMX3DUE
PUBLIC MDTASIDUE,MDTLOFDUE,MDATEACPT,MDATEWARX,MMILE1DUE,MMILE2DUE
PUBLIC MMILE3DUE,MMILEASI,MMILEWARX,MMILELOF,MVISNUM,MNOUN,MASC
PUBLIC MVORGCODE,MNUMAUTH,MNUMASGN,MMISESSN,MVALMGT,MMGTISORG
PUBLIC MMGTISASC,MPRIREC,MMGTCODE,MASGNMGT,MISNUM,MNSN,MASGNIS
PUBLIC MNOMEN,MORGCODE,MOWNCMD,MUSECMD,MITEMCODE,MREPCODE,MUSECODE
PUBLIC MSPEQCODE,MDATEASGN,MMOBLCODE,MDATEDUE,MPKLOTNO,MPKROWNO
PUBLIC MPKCOLNO,MPKSTATUS,MREAPLVL,MRESTAT,MREORG,MRENAMEPH,MRERQDATE
PUBLIC MREACTDT,MREESTRTN,MRECASEN,MRECATEG,MRERMKS,MMXSTATUS,MMXRTNDATE
PUBLIC MMXRMKS,MSIINTERVAL,MSISWITCH,MLOFINTER,MLOFSWITCH,MEXLSTDT
PUBLIC MEXNXTDT,MEXRMKS,MDISPDEST,MDISPRQDT,MDISPMHRS,MDISRMK1,MDISINST
PUBLIC SWITCH,MTITLE,REPEAT,MBDISPDT,MEDISPDT,OWRITE,MBEGINDT,MENDDT
*************************************************************************
***ITITIALIZE VARIABLES***
*****************************
STORE SPACE(8) TO MBEGINDT
STORE SPACE(8) TO MENDDT
STORE  0 TO MBDISPDT
STORE  0 TO MEDISPDT
STORE  "N" TO OWRITE
STORE "Y" TO REPEAT
STORE "WRM VEHICLE MANAGEMENT SYSTEM" TO MTITLE
STORE 0 TO SWITCH
STORE SPACE(8) TO MREGNUM
STORE 0 TO MTOTMILAGE
STORE SPACE(1) TO MVEHTYPE
STORE SPACE(8) TO MDTMX1DUE
STORE SPACE(8) TO MDTMX2DUE
STORE SPACE(8) TO MDTMX3DUE
STORE SPACE(8) TO MDTASIDUE
```

```
STORE SPACE(8) TO MDTLOFDUE
STORE SPACE(8) TO MDATEACPT
STORE SPACE(8) TO MDATEWARX
STORE 0 TO MMILE1DUE
STORE 0 TO MMILE2DUE
STORE 0 TO MMILE3DUE
STORE 0 TO MMILEASI
STORE 0 TO MMILEWARX
STORE 0 TO MMILELOF
STORE SPACE(18) TO MVISNUM
STORE SPACE(25) TO MNOUN
STORE SPACE(7)  TO MASC
STORE SPACE(2)  TO MVORGCODE
STORE  0 TO MNUMAUTH
STORE  0 TO MNUMASGN
STORE  0 TO MMISESSN
STORE SPACE(4)  TO  MVALMGT
STORE SPACE(31) TO  MMGTISORG
STORE SPACE(29) TO  MMGTISASC
STORE  0 TO MPRIREC
STORE SPACE(4)  TO MMGTCODE
STORE SPACE(4)  TO MASGNMGT
STORE SPACE(18) TO MISNUM
STORE SPACE(18) TO MNSN
STORE SPACE(18) TO MASGNIS
STORE SPACE(25) TO MNOMEN
STORE SPACE(2)  TO MORGCODE
STORE SPACE(2)  TO MOWNCMD
STORE SPACE(2)  TO MUSECMD
STORE SPACE(1)  TO MITEMCODE
STORE SPACE(1)  TO MREPCODE
STORE SPACE(1)  TO MUSECODE
STORE SPACE(10) TO MSPEQCODE
STORE SPACE(8)  TO MDATEASGN
STORE SPACE(1)  TO MMOBLCODE
STORE SPACE(8)  TO MDATEDUE
STORE SPACE(7)  TO MASC
STORE 0 TO MPKLOTNO
STORE 0 TO MPKROWNO
STORE 0 TO MPKCOLNO
STORE SPACE(1) TO  MPKSTATUS
STORE SPACE(1) TO  MREAPLVL
STORE SPACE(1) TO  MRESTAT
STORE SPACE(2) TO  MREORG
STORE SPACE(25) TO MRENAMEPH
STORE SPACE(8)  TO MRERQDATE
STORE SPACE(8)  TO MREACTDT
STORE SPACE(8)  TO MREESTRTN
STORE SPACE(7)  TO  MRECASEN
STORE SPACE(1)  TO  MRECATEG
STORE SPACE(50) TO  MRERMKS
STORE SPACE(1)  TO  MMXSTATUS
STORE SPACE(8)  TO MMXRTNDATE
STORE SPACE(50) TO  MMXRMKS
```

```
STORE 0 TO  MSIINTERVAL
STORE SPACE(1) TO MSISWITCH
STORE 0 TO  MLOFINTER
STORE SPACE(1) TO MLOFSWITCH
STORE SPACE(8) TO MEXLSTDT
STORE SPACE(8) TO MEXNXTDT
STORE SPACE(50) TO MEXRMKS
STORE SPACE(25) TO MDISPDEST
STORE 0 TO MDISPRQDT
STORE 0 TO MDISPMHRS
STORE SPACE(50) TO MDISRMK1
STORE SPACE(50) TO MDISINST
**************************
***LOAD PROCEDURE FILE***
**************************
SET PROCEDURE TO PROCA
************************************
***RESTRICTED RIGHTS WARNING SCREEN
************************************
@ 0,0 Clear
@ 1,0  to 21,78 double
    @ 4,25 say "WRM VEHICLE MANAGEMENT SYSTEM"
    @ 7,19 say "WRM Vehicle Management System version 1.0"
    @ 9,12 say " Copyright (c) Robert S. Thomas 1988.  All Rights
                Reserved"
    @ 11,21 say " ******Restricted Rights Warning******"
    @ 13,11 say " The WRM Vehicle Management System is a copyrighted
                package"
    @ 14,7  say " designed for the exclusive use of the United States
                Military, and"
    @ 15,7  say " is protected by U.S. Copyright Law (Title 17 United
                States Code)."
    @ 16,6  say " Unauthorized reproduction and / or sales may result in
                 imprisonment"
    @ 17,12 say " of up to ONE YEAR and FINES up to $10,000.(17 USC 506)"
    @ 18,10 say " Copyright infringers may also be subject to Civil
                Liability."
    @ 20,12 say " Copyright (c) Robert S. Thomas 1988. All Rights Reserved"
    @ 23,23 say "****PRESS ANY KEY TO CONTINUE****"
            wait ""
**************************
***DEFINE STARTUP MENU***
**************************
    @ 0,0 clear
     store  0 to choice
  do while .t.
    @ 0,0 clear
    @ 5,18 to 17,63 double
    @ 7,19 to 7,62 double
    @ 6,26 SAY "WRM VEHICLE MANAGEMENT SYSTEM"
    @ 9,28 SAY "1. WRM SYSTEM MAIN MENU"
    @ 11,28 say "2. WRM SYSTEM UTILITIES"
    @ 13,28 say "0. EXIT THE SYSTEM TO DOS"
    @ 16,33 say "MAKE A CHOICE" get choice picture "9"
```

```
     read
*********************
***ACCEPT CHOICES***
*********************
      Do Case
         Case choice = 1
            @ 0,0 clear
              Do main
         CASE CHOICE = 2
            @ 0,0 CLEAR
              DO uTIL
         Case choice = 0
              quit
      Endcase
Enddo
                              *************************
                              *       MAIN.PRG        *
                              *diverts to system prgms*
                              *          or           *
                              *diverts to system utils*
                              *************************

*********************************
PROGRAMMER ROBERT S THOMAS 1988*
CALLS: DBRTNS.PRG, PKDIS.PRG   *
       SCHEDACT.PRG,RELRTNS.PRG*
       RPTRTNS.PRG             *
*********************************
***SET ENVIRONMENT****
***********************
set talk off
set escape off
clear gets
 @ 0,0 clear
  store 0 to choice
  DO WHILE .T.
***********************
***DEFINE MAIN MENU***
***********************
 @ 5,15 to 22,70 double
 @ 8,16 to 8,69 double
      @ 6,28 say "WRM Vehicle Management System"
      @ 7,38 say "Main Menu"
      @ 9,24 say "1. Database File Manipulation Routines"
      @ 11,24 say "2. Inventory/Storage Dispersal Management"
      @ 13,24 say "3. Scheduled Actions Management"
      @ 15,24 say "4. Release Case Management"
      @ 17,24 say "5. Reports Menu"
      @ 19,24 say "0. Quit the System"
 @ 21,50 to 23,66 double
***********************
***GET CHOICES***
*****************
 @ 22,51 say "Make a Choice "
```

```
● 22,65 GET CHOICE PICTURE "9"
read
● 0,0 clear
   do case
      case choice = 1
        clear gets
        ● 0,0 clear
        DO DBRTNS
      case choice = 2
        CLEAR GETS
        ● 0,0 CLEAR
        DO PKDIS
      case choice = 3
        clear gets
        ● 0,0 clear
        DO SCHEDACT
      case choice = 4
        clear gets
        ● 0,0 clear
        DO RELRTNS
      case choice = 5
        clear gets
        ● 0,0 clear
        DO RPTRTNS
      case choice = 0
         store 0 to choice
         ● 0,0 clear
         RETURN TO MASTER
   endcase
ENDDO
```

```
                          ***************************
                          *        DBRTNS.PRG       *
                          *  add/edit/view/delete   *
                          *  undelete from dbases   *
                          ***************************

*********************************
PROGRAMMER: ROBERT S THOMAS 1988
     CALLS: PROCA.PRG, MAIN.PRG
*********************************
***ENVIRONMENT***
*****************
SET SAFETY OFF
● 0,0 clear
*****************
***DEFINE MENU***
*************************************************************************
● 0,7 say "****THE DATABASES CONTAIN THE FOLLOWING FIELDS OF
          INFORMATION****"
●  1,0 TO 24,79 DOUBLE
●  3,1 TO 3,78 DOUBLE
● 21,1 TO 21,78 DOUBLE
●  2,1 SAY "    MASTER        OLVIMS        VAL      RELEASE
```

```
            MANMAINT      DISPERSAL"
● 4,1 SAY "     REGNUM         REGNUM       REGNUM      REGNUM
            REGNUM        REGNUM"
● 5,1 SAY "    MGTCODE     TOTMILEAGE       VISNUM     REAPLVL
            MXSTATUS      DISPDEST"
● 6,1 SAY "   ASGN:MGT        VEHTYPE         NOUN      RESTAT
            MXRTNDATE      DISPRQDT"
● 7,1 SAY "     IS:NUM       DTMX1DUE          ASC       REORG
            MXRMKS      DISPMHRS"
● 8,1 SAY "         NSN       DTMX2DUE     VORGCODE    RENAMEPH
            SIINTERVAL      DISPINST"
● 9,1 SAY "    ASGN:IS        DTMX3DUE     NUM:AUTH    RERQDATE
            SISWITCH"
● 10,1 SAY "      NOMEN        DTASIDUE     NUM:ASGN     REACTDT
            LOFINTER"
● 11,1 SAY "    ORGCODE        DTLOFDUE     MIS:ESSN    REESTRTN
            LOFSWITCH"
● 12,1 SAY "     OWNCMD        DATEACPT       VALMGT     RECASEN"
● 13,1 SAY "     USECMD        DATEWARX     MGTISORG     RECATEG"
● 14,1 SAY "   ITEMCODE        MILE1DUE     MGTISASC      RERMKS"
● 15,1 SAY "    REPCODE        MILE2DUE      PRI:REC"
● 16,1 SAY "    USECODE        MILE3DUE"
● 17,1 SAY "   SPEQCODE         MILEASI"
● 18,1 SAY "  DATE:ASGN        MILEWARX"
● 19,1 SAY "   MOBLCODE         MILELOF"
● 20,1 SAY "   DATE:DUE"
● 20,1 SAY "        ASC"
● 13,52 TO 15,77 DOUBLE
● 14,53 SAY "   PKPLAN     EXERCISE"
● 16,53 SAY "   REGNUM       REGNUM"
● 17,53 SAY "  PKLOTNO      EXLSTDT"
● 18,53 SAY "  PKROWNO      EXNXTDT"
● 19,53 SAY "  PKCOLNO       EXRMKS"
● 20,53 SAY " PKSTATUS"
● 22,7 SAY "**SELECT THE CAPITALIZED LETTER OF YOUR CHOICE AND
           PRESS RETURN**"
● 23,1 SAY " [M]aster [O]lvims [V]al [R]elease m[A]nmaint
           [D]ispersal [P]kplan [E]xercise"
● 17,34 TO 19,45
*********************************************************************
***GET CHOICES***
*****************
● 18,35 SAY "CHOICE ? "
  STORE SPACE(1) TO SELECTION
● 18,44 GET SELECTION PICTURE "●! A"
  READ
  DO CASE
     CASE SELECTION = "M"
          STORE 1 TO SWITCH
     CASE SELECTION = "O"
          STORE 2 TO SWITCH
     CASE SELECTION = "V"
          STORE 3 TO SWITCH
     CASE SELECTION = "R"
```

```
          STORE 4 TO SWITCH
     CASE SELECTION = "A"
          STORE 5 TO SWITCH
     CASE SELECTION = "D"
          STORE 6 TO SWITCH
     CASE SELECTION = "P"
          STORE 7 TO SWITCH
     CASE SELECTION = "E"
          STORE 8 TO SWITCH
  ENDCASE
*******************************
***DEFINE DATABASE SUMMARY***
*******************************
● 0,0 CLEAR
● 0,1 TO 24,78 DOUBLE
● 3,2 TO 3,77 DOUBLE
● 21,2 TO 21,77 DOUBLE
● 1,32 SAY  "DATABASE SUMMARY"
  DO HEADER
● 22,7 SAY " **SELECT THE CAPITALIZED LETTER OF YOUR CHOICE
           AND PRESS RETURN**"
● 23,2 SAY "     "
● 23,11 SAY "[H]elp  [V]iew  [A]dd  [E]dit  [D]elete
             [U]ndelete  [Q]uit "
  STORE  "Q" TO MSELECT
● 17,34 TO 19,45 DOUBLE
● 18,35 SAY "CHOICE ?"
● 18,44 GET MSELECT PICTURE "@! A"
    READ
         DO CASE
********************
***HELP ROUTINES***
********************
            CASE MSELECT = "H"
                 IF SWITCH = 1
                    DO HMAST
                 ENDIF
                 IF SWITCH = 2
                    DO HVAL
                 ENDIF
                 IF SWITCH = 3
                    DO HVIMS
                 ENDIF
                 IF SWITCH = 4
                    DO HREL
                 ENDIF
                 IF SWITCH = 5
                    DO HMAN
                 ENDIF
                 IF SWITCH = 6
                    DO HDIS
                 ENDIF
                 IF SWITCH = 7
                    DO HPARK
```

```
                ENDIF
                IF SWITCH = 8
                   DO HEXE
                ENDIF
*******************
***VIEW ROUTINES***
*******************
            CASE MSELECT = "V"
             DO WHILE .T.
                DO SEARCHER
                STORE "VIEW" TO NOUNS
                DO SHOWSWITCH
                DO REPEATER
                LOOP
              ENDDO
*******************
***ADD ROUTINES***
*******************
            CASE MSELECT = "A"
              STORE "ADD" TO NOUNS
               DO WHILE .T.
                    IF SWITCH = 1
                       USE MASTER INDEX REGMAST
                       DO GETMAST
                    ENDIF
                    IF SWITCH = 2
                       USE VIMS INDEX REGVIMS
                       DO GETVIMS
                    ENDIF
                    IF SWITCH = 3
                       USE VAL INDEX REGVAL
                       DO GETVAL
                    ENDIF
                    IF SWITCH = 4
                       USE RELEASE INDEX REGREL
                       DO GETREL
                    ENDIF
                    IF SWITCH = 5
                       USE MANMAINT INDEX REGMAN
                       DO GETMAN
                    ENDIF
                    IF SWITCH = 6
                       USE DISDIS INDEX REGDIS
                       DO GETDIS
                    ENDIF
                    IF SWITCH = 7
                       USE PKPLAN INDEX REGPARK
                       DO GETPARK
                    ENDIF
                    IF SWITCH = 8
                       USE EXERCISE INDEX REGEXE
                       DO GETEXE
                    ENDIF
                  DO REPEATER
```

```
                LOOP
              ENDDO

*********************
***EDIT ROUTINES***
*********************
  CASE MSELECT = "E"
             STORE "EDIT" TO NOUNS
              DO WHILE .T.
                DO SEARCHER
                   IF SWITCH = 1
                      DO STOREMAST
                      DO GETMAST
                   ENDIF
                   IF SWITCH = 2
                      DO STOREVIMS
                      DO GETVIMS
                   ENDIF
                   IF SWITCH = 3
                      DO STOREVAL
                      DO GETVAL
                   ENDIF
                   IF SWITCH = 4
                      DO STOREREL
                      DO GETREL
                   ENDIF
                   IF SWITCH = 5
                      DO STOREMAN
                      DO GETMAN
                   ENDIF
                   IF SWITCH = 6
                      DO STOREDIS
                      DO GETDIS
                   ENDIF
                   IF SWITCH = 7
                      DO STOREPARK
                      DO GETPARK
                   ENDIF
                   IF SWITCH = 8
                      DO STOREEXE
                      DO GETEXE
                   ENDIF
                DO REPEATER
                LOOP
              ENDDO
**********************
***DELETE ROUTINES***
**********************
           CASE MSELECT = "D"
             STORE "DELETE" TO NOUNS
              DO WHILE .T.
                DO SEARCHER
                   IF SWITCH = 1
                      DO SHOWMAST
```

```
                    ENDIF
                    IF SWITCH = 2
                       DO SHOWVIMS
                    ENDIF
                    IF SWITCH = 3
                       DO SHOWVAL
                    ENDIF
                    IF SWITCH = 4
                       DO SHOWREL
                    ENDIF
                    IF SWITCH = 5
                       DO SHOWMAN
                    ENDIF
                    IF SWITCH = 6
                       DO SHOWDIS
                    ENDIF
                    IF SWITCH = 7
                       DO SHOWPARK
                    ENDIF
                    IF SWITCH = 8
                       DO SHOWEXE
                    ENDIF
                  DO DELETER
                  DO REPEATER
                  LOOP
                ENDDO
************************
***UNDELETE ROUTINES***
************************
            CASE MSELECT = "U"
              STORE "UNDELETE" TO NOUNS
               DO WHILE .T.
                 DO SWITCHER
                 DO SEARCHER
                 DO SHOWSWITCH
                 DO UNDELETER
                 DO REPEATER
                 LOOP
               ENDDO
            CASE MSELECT = "Q"
                 CLOSE DATABASES
                 CLEAR GETS
                 DO MAIN
      ENDCASE

                        ****************************
                        *       SCHEDACT.PRG       *
                        *schedued actions module *
                        ****************************
*********************************
PROGRAMMER: ROBERT S THOMAS 1988
     CALLS: SCHEDPRO.PRG
*********************************
STORE 0 TO CHOICE
```

```
● 0,0 CLEAR
● 2,13 TO 23,67 DOUBLE
● 3,25 SAY MTITLE
**********************************
***DEFINE SCHEDULED ACTIONS MENU***
**********************************
● 4,29  SAY "SCHEDULED ACTIONS MENU"
● 5,14 TO 5,66 DOUBLE
● 7,15  SAY "1. View all scheduled actions for a vehicle"
● 9,15  SAY "2. View all vehicles scheduled for maintenance"
● 10,15 SAY "   action for a user selected time period"
● 12,15 SAY "3. View all vehicles schedued for exercise
● 13,15 SAY "   for a user selected time period
● 15,15 SAY "0. Return to main menu"
● 16,46 TO 18,63 DOUBLE
● 17,47 SAY " MAKE A CHOICE "
********************
***MAKE A CHOICE***
********************
● 17,62 GET CHOICE PICTURE "#"
  READ
  DO CASE
*************************
***VIEW ALL SCHEDULED***
*************************
     CASE CHOICE = 1
       DO SCHEDPRO
****************************
***VIEW ALL SCHED MAINT***
****************************
    CASE CHOICE = 2
       DO WHILE .T.
         STORE 9 TO SWITCH
         USE VIMS INDEX REGVIMS
         ● 0,0 CLEAR
           DO HEADER
           DO DTGET
           DO PRINTER
           DO EOFTEST
           DO AGAIN
           LOOP
       ENDDO
**********************************
***VIEW ALL SCHED FOR EXERCISE***
**********************************
     CASE CHOICE = 3
       DO WHILE .T.
         STORE 10 TO SWITCH
         USE VIMS INDEX REGVIMS
         ● 0,0 CLEAR
           DO HEADER
           DO DTGET
           DO PRINTER
           DO EOFTEST
```

```
          DO AGAIN
          LOOP
        ENDDO
      CASE CHOICE = 0
        RETURN
      ENDCASE


                              **************************
                              *       SCHEDPRO.PRG     *
                              *displays all scheduled *
                              *    vehicle actions     *
                              **************************

**********************************
PROGRAMMER: ROBERT S THOMAS 1988
     CALLS: MAIN.PRG
**********************************
DO WHILE .T.
● 0,0 clear
*******************************
***GET VEHICLE REG. NUMBER***
*******************************
● 2,19 TO 4,61 DOUBLE
● 3,20 SAY "DISPLAY SCHEDULED ACTIONS FOR A VEHICLE"
● 12,16 SAY "ENTER THE VEHICLE REGISTRATION NUMBER "
● 11,10 TO 13,65 DOUBLE
  STORE SPACE(8) TO REGNUM
● 12,55 GET REGNUM PICTURE "@! 99A99999"
  READ
*****************************
***SET DATABASE REATIONS***
*****************************
   SELECT  1
     USE PKPLAN INDEX REGPARK
   SELECT  2
     USE EXERCISE INDEX REGEXE
     SET RELATION TO REGNUM INTO PKPLAN
   SELECT  3
     USE RELEASE INDEX REGREL
     SET RELATION TO REGNUM INTO EXERCISE
   SELECT  4
     USE MANMAINT INDEX REGMAN
     SET RELATION TO REGNUM INTO RELEASE
   SELECT  5
     USE VIMS INDEX REGVIMS
     SET RELATION TO REGNUM INTO MANMAINT
   SELECT  6
     USE MASTER INDEX REGMAST
     SET RELATION TO REGNUM INTO VIMS
   SELECT MASTER
*******************************
***FIND VEHICLE REG. NUMBER***
*******************************
```

```
        GO TOP
        FIND &REGNUM
          STORE MASTER->MGTCODE  TO MMGTCODE
          STORE MASTER->SPEQCODE TO MSPEQCODE
          STORE MASTER->MOBLCODE TO MMOBLCODE
          STORE MASTER->ASC      TO MASC
          STORE MASTER->DATEASGN TO MDATEASGN
          STORE VIMS->VEHTYPE    TO MVEHTYPE
          STORE VIMS->DATEACPT   TO MDATEACPT
          STORE VIMS->DATEWARX   TO MDATEWARX
          STORE VIMS->DTMX1DUE   TO MDTMX1DUE
          STORE VIMS->DTMX2DUE   TO MDTMX2DUE
          STORE VIMS->DTMX3DUE   TO MDTMX3DUE
          STORE VIMS->DTASIDUE   TO MDTASIDUE
          STORE VIMS->DATELOFDUE TO MDTLOFDUE
          STORE VIMS->TOTMILEAGE TO MTOTMILAGE
          STORE VIMS->MILE1DUE   TO MMILE1DUE
          STORE VIMS->MILE2DUE   TO MMILE2DUE
          STORE VIMS->MILE3DUE   TO MMILE3DUE
          STORE VIMS->MILEASI    TO MMILEASI
          STORE VIMS->MILELOF    TO MMILELOF
          STORE VIMS->MILEWARX   TO MMILEWARX
          STORE MANMAINT->MXSTATUS   TO MMXTATUS
          STORE MANMAINT->MXRTNDATE  TO MMXRTNDATE
          STORE MANMAINT->MXRMKS     TO MMXRMKS
          STORE MANMAINT->SIINTERVAL TO MSIINTERVAL
          STORE MANMAINT->SISWITCH   TO MSISWITCH
          STORE MANMAINT->LOFINTERV  TO MLOFINTER
          STORE MANMAINT->LOFSWITCH  TO MLOFSWITCH
          STORE EXERCISE->EXLSTDT TO MEXLSTDT
          STORE EXERCISE->EXNXTDT TO MEXNXTDT
          STORE EXERCISE->EXRMKS  TO MEXRMKS
          STORE RELEASE->RESTAT   TO MRESTAT
          STORE RELEASE->RERQDATE TO MRERQDATE
          STORE RELEASE->REACTDT  TO MREACTDT
          STORE RELEASE->REESTRTN TO MREESTRTN
          STORE RELEASE->RECASEN  TO MRECASEN
          STORE PKPLAN->PKLOTNO  TO MPKLOTNO
          STORE PKPLAN->PKROWNO  TO MPKROWNO
          STORE PKPLAN->PKCOLNO  TO MPKCOLNO
          STORE PKPLAN->PKSTATUS TO MPKSTATUS
*****************************
***DISPLAY TO THE SCREEN***
*****************************
@ 0,0 CLEAR
@ 1,0 TO 6,79 DOUBLE
@ 0,1 SAY "SCHEDULED ACTIONS REPORT FOR VEHICLE " +REGNUM
@ 0,50 SAY "PK LOT# "+ LTRIM(STR(MPKLOTNO))
@ 0,62 SAY "ROW " + LTRIM(STR(MPKROWNO))
@ 0,70 SAY "COL " + LTRIM(STR(MPKCOLNO))
@ 2,2  SAY "TOTAL VEHICLE LIFETIME MILEAGE " +
           LTRIM(STR(MTOTMILAGE))
@ 3,2  SAY "    DATE ACCEPTED BY AIR FORCE " + DTOC(MDATEACPT)
@ 4,2  SAY "    DATE ASSIGNED TO WRM FLEET " + DTOC(MDATEASGN)
```

```
@ 5,2  SAY " WRM IDENTIFIER CODE FROM VIMS " + MMOBLCODE
@ 2,44 SAY "VEHICLE MANAGEMENT CODE " + MMGTCODE
@ 3,44 SAY " SPECIAL EQUIPMENT CODE " + MSPEQCODE
@ 4,44 SAY "  ALLOWANCE SOURCE CODE " + MASC
@ 5,44 SAY "           VEHICLE TYPE " + MVEHTYPE
@ 7,8 SAY "RELEASE CASE INFORMATION"
@ 8,0 to 14,41
@ 9,2 SAY "        CURRENT RELEASE STATUS "  + MRESTAT
@ 10,2 SAY " CURRENT/LAST RELEASE CASE NO. " + MRECASEN
@ 11,2 SAY "  DATE VEHICLE RELEASE DESIRED " + DTOC(MRERQDATE)
@ 12,2 SAY "DATE VEHICLE ACTUALLY RELEASED " + DTOC(MREACTDT)
@ 13,2 SAY "  ESTIMATED/ACTUAL RETURN DATE " + DTOC(MREESTRTN)
@ 15,10 SAY "EXERCISE INFORMATION"
@ 16,0  to 19,41
@ 17,2  SAY "   DATE VEHICLE LAST EXERCISED " + DTOC(MEXLSTDT)
@ 18,2  SAY " DATE VEHICLE DUE FOR EXERCISE " + DTOC(MEXNXTDT)
@ 21,0 to 24,79
@ 24,26  SAY "TO CONTINUE PRESS ANY KEY !"
@ 20,30 SAY "REMARKS"
@ 22,1  SAY "CURRENT MAINTENANCE STATUS  " + MMXRMKS
@ 23,1  SAY "   CURRENT EXERCISE STATUS  " + MEXRMKS
@ 7,48  SAY "MAINTENANCE INFORMATION"
@ 8,42 to 20,79
@ 9,71 TO 16,71
@  9,63 SAY "DATE DUE"
@  9,72 SAY "MILEAGE"
@ 10,43 SAY "SCHEDULED MAINT. #1 " + DTOC(MDTMX1DUE)
@ 11,43 SAY "SCHEDULED MAINT. #2 " + DTOC(MDTMX2DUE)
@ 12,43 SAY "SCHEDULED MAINT. #3 " + DTOC(MDTMX3DUE)
@ 10,72 SAY LTRIM(STR(MMILE1DUE))
@ 11,72 SAY LTRIM(STR(MMILE2DUE))
@ 12,72 SAY LTRIM(STR(MMILE3DUE))
@ 13,43 SAY "ANNUAL SAFETY INSP. " + DTOC(MDTASIDUE)
@ 17,43 TO 17,78
@ 14,43 SAY "SCHEDULED ON MONTHS IF YES " + MSISWITCH
@ 14,72 SAY "#MON " + LTRIM(STR(MSIINTERVAL))
@ 15,43 SAY "LUBE & OIL & FILTER " + DTOC(MDTLOFDUE)
@ 16,43 SAY "SCHEDULED ON MONTHS IF YES " + MLOFSWITCH
@ 16,72 SAY "#MON " + LTRIM(STR(MLOFINTER))
@ 18,43 SAY "CURRENT MAINT. STATUS " + MMXSTATUS
@ 19,43 SAY "        DATE DUE BACK " + DTOC(MMXRTNDATE)
  WAIT ""
   DO AGAIN
   DO SCHEDPRO
ENDDO
```

```
**********************
*      PKDIS.PRG     *
* Inventory display  *
* Dispersal display  *
**********************
```

```
 *********************************
 PROGRAMMER: ROBERT S THOMAS 1988
      CALLS: APARK.PRG
 *********************************
● 0,0 CLEAR
  CLEAR GETS
● 2,14 TO 20,66 DOUBLE
● 3,25 SAY MTITLE
● 4,26 SAY "Dispersal/Distribution Menu"
● 5,15 to 5,65 double
●  7,20 SAY "1. Display all vehicles by Park Lot Number"
●  9,20 SAY "2. Display vehicles by park status"
● 11,20 SAY "3. Produce automated park plan"
● 13,20 SAY "4. Display dispersal list by date required"
● 15,20 SAY "5. Display dispersal list by destination"
● 17,20 SAY "0. Return to Main Menu"
● 19,50 TO 21,63 DOUBLE
● 20,51 SAY " Choice ?  "
  STORE "0" TO ANSWER
● 20,62 GET ANSWER PICTURE "9"
  READ
DO CASE
 *********************************
 ***DISPLAY ALL BY PK LOT NO.***
 *********************************
   CASE ANSWER = "1"
      DO WHILE .T.
         ● 0,0 CLEAR
         STORE 1 TO SWITCH
         USE PKPLAN INDEX PKLTNUM
         GO TOP
         ● 0,0 CLEAR
         ● 12,24 SAY "ENTER THE PARKING LOT NUMBER ? "
         ● 12,57 GET MPKLOTNO PICTURE "999"
           READ
           DO PRINTER
           DO EOFTEST
           DO AGAIN
           LOOP
      ENDDO
 ***************************
 ***DISPLAY BY PK STATUS***
 ***************************
  CASE ANSWER = "2"
  ● 0,0 CLEAR
   DO WHILE .T.
      STORE 2 TO SWITCH
      USE PKPLAN
       ● 0,0 CLEAR
       ● 12,20 SAY "ENTER THE PARKING STATUS YOU WISH TO SEE"
       ● 13,29 SAY "A LIST OF VEHICLES FOR"
       ● 13,52 GET MPKSTATUS PICTURE "@! A"
         READ
         DO PRINTER
```

```
        DO EOFTEST
        DO AGAIN
        LOOP
  ENDDO
**************************
***PRODUCE AUTO PK PLAN***
**************************
CASE ANSWER = "3"
        DO APARK
        DO AGAIN
        LOOP
**************************
***DISPERSAL BY DT. RQD***
**************************
CASE ANSWER = "4"
  DO WHILE .T.
   STORE 3 TO SWITCH
   USE DISDIS
   @  0,0 CLEAR
   @  1,1 TO 19,78 DOUBLE
   @  2,5 SAY "****THE FORMAT USED FOR DATE IS BASED UPON D +
              (NO DAYS) NOTATION****"
   @  4,5 SAY "THIS FORM OF NOTATION PERMITS INPUT INTO THE
              SYSTEM WITHOUT REGARD TO"
    @  5,5 SAY "DATES OR MENTION OF WARPLANS OR EXERCISE
              NAMES. THIS FORM OF SCHEDULING"
    @  6,5 SAY "PERMITS THE DISPERSAL MODULE TO BE USED FOR A
              WIDE RANGE OF PLANNING."
    @  8,8 SAY " ***    **      *  *   **  *****    ***  *  *
              ***   *  * *****"
    @  9,8 SAY " *  *  *  *     ** *  *  *   *       *   ** *
              *  *  *  *   *  "
    @ 10,8 SAY " *  *  *  *     * **  *  *   *       *   * **
              ***   *  *   *  "
    @ 11,8 SAY " ***    **      *  *   **    *      ***  *  *
              *      **    *  "
    @ 13,5 SAY "  ***   *       **     **     **    ****
              ****   ****   ****   *** "
    @ 14,5 SAY " *      *      *  *   *  *   *  *    **     *
              **    *      *  *"
    @ 15,5 SAY " *      *      *  *    *      *      **     *
              **    *      *  *"
    @ 16,5 SAY " *      *      ****     *      *     **
              ***     **    ***    *  *"
    @ 17,5 SAY " *      *      *  *   *  *   *  *    **     *
              **    *      *  *"
    @ 18,5 SAY "  ***   ****   *  *    **     **    ****    *
              ****   ****   *** "
   @ 20,20 SAY "ENTER THE BEGINNING DISPERSAL DATE"
     @ 20,54 GET MBDISPDT PICTURE "999"
    @ 21,23 SAY "ENTER THE ENDING DISPERSAL DATE"
     @ 21,54 GET MEDISPDT PICTURE "999"
    READ
    DO PRINTER
```

```
     DO EOFTEST
     DO AGAIN
     LOOP
   ENDDO
*******************************
***DISPERSAL BY DESTINATION***
*******************************
 CASE ANSWER = "5"
   DO WHILE .T.
    STORE 4 TO SWITCH
    USE DISDIS
     @ 0,0 CLEAR
     @ 12,27 SAY "ENTER THE DESTINATION CODE"
       @ 12,40 GET MDISPDEST PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXXX"
     READ
     DO PRINTER
     DO EOFTEST
     DO AGAIN
     LOOP
   ENDDO
 CASE ANSWER = "0"
    @ 0,0 CLEAR
    DO MAIN
 ENDCASE

                              **************************
                              *      RELRTNS.PRG       *
                              *  release dbase rtns    *
                              **************************

*********************************
PROGRAMMER: ROBERT S THOMAS 1988
     CALLS:
*********************************
@  0,0 CLEAR
    CLEAR GETS
*********************************
***DEFINE RELEASE MENU SCREEN***
*********************************
@  2,16 TO 18,64 DOUBLE
@  5,17 TO 5,63 DOUBLE
@  3,25 SAY MTITLE
@  4,31 SAY "RELEASE CASE MENU"
@  7,18 SAY "1. Display All Open Release Cases"
@  9,18 SAY "2. Display Release Cases By Release Category"
@ 11,18 SAY "3. Display Release Cases By Organization Code"
@ 13,18 SAY "4. Display Individual Vehicle Release Case"
@ 15,18 SAY "0. Return To Main Menu"
@ 17,48 TO 19,59 DOUBLE
@ 18,49 SAY "Choice ? "
    STORE 0 TO CHOICE
****************
***GET CHOICE***
```

```
****************
  @ 18,58 GET CHOICE PICTURE "9"
    READ
DO CASE
**************************
***DISPLAY ALL OPEN REL***
**************************
  CASE CHOICE = 1
    DO WHILE .T.
      STORE 13 TO SWITCH
      @ 0,0 CLEAR
        USE RELEASE INDEX REGREL
        DO PRINTER
        DO EOFTEST
        DO AGAIN
        LOOP

    ENDDO
**************************
***DISPLAY BY REL CATEG***
**************************
  CASE CHOICE = 2
    DO WHILE .T.
      STORE 14 TO SWITCH
      @ 0,0 CLEAR
        USE RELEASE INDEX REGREL
      @ 12,24 SAY "ENTER THE RELEASE CASE CATEGORY"
      @ 12,57 GET MRECATEG PICTURE "@! X"
        READ
        DO PRINTER
        DO EOFTEST
        DO AGAIN
        LOOP
    ENDDO
************************
***DISPLAY REL BY ORG***
************************
 CASE CHOICE = 3
    DO WHILE .T.
      STORE 15 TO SWITCH
      @ 0,0 CLEAR
        USE RELEASE INDEX REGREL
      @ 12,25 SAY "ENTER THE ORGANIZATION CODE"
      @ 12,56 GET MREORG PICTURE "@! XX"
        READ
        DO PRINTER
        DO EOFTEST
        DO AGAIN
        LOOP
    ENDDO
************************
***VIEW A VEHICLE REL***
************************
  CASE CHOICE = 4
```

```
    DO WHILE .T.
      STORE 16 TO SWITCH
      @ 0,0 CLEAR
        USE RELEASE INDEX REGREL
        DO SEARCHER
        DO SHOWREL
        DO REPEATER
        LOOP
    ENDDO
  CASE CHOICE = 0
    @ 0,0 CLEAR
    RETURN
  ENDCASE

                              **********************
                              *     RPTRTNS.PRG    *
                              *  produces reports  *
                              **********************

**********************************
PROGRAMMER: ROBERT S THOMAS 1988
     CALLS: REPORT FORMS
**********************************
@ 0,0 CLEAR
@ 0,10 TO 24,70 DOUBLE
**************************
***DEFINE REPORT MENU***
**************************
@ 1,25 SAY MTITLE
@ 2,34 SAY "REPORTS MENU"
@ 3,11 TO 3,69 DOUBLE
@ 22,11 TO 22,69 DOUBLE
@  5,19 SAY "1. VEHICLE AUTHORIZATIONS/ASSIGNMENTS REPORT"
@  7,19 SAY "2. WRM VEHICLE VAL REPORT"
@  9,19 SAY "3. STORAGE/DISPERSAL/CAPABILITY REPORT"
@ 11,19 SAY "4. DISPERSAL CHECKLIST"
@ 13,19 SAY "5. SCHEDULING REPORT"
@ 15,19 SAY "6. SCHEDULING CHECKLIST"
@ 17,19 SAY "7. CURRENT RELEASE CASE STATUS REPORT"
@ 19,19 SAY "8. RELEASE CASE ANALYSIS REPORT"
@ 21,19 SAY "0. RETURN TO MAIN MENU"
  CLEAR GETS
     STORE 0 TO CHOICE
     @ 23,35 SAY "MAKE A CHOICE ?"
     @ 23,51 GET CHOICE PICTURE "9"
    READ
 DO CASE
**************************
***VEH AUTH/ASSGN RPT***
**************************
  CASE CHOICE = 1
   DO WHILE .T.
      USE MASTER INDEX REGMAST
         SET PRINT ON
```

```
            ? CHR(27) + CHR(15)
            ? CHR(27) + CHR(18)
          SET PRINT OFF
        CLOSE DATABASES
        DO AGAIN
        LOOP
    ENDDO
**********************
***WRM VEH VAL RPT***
**********************
  CASE CHOICE = 2
   DO WHILE .T.
      USE VAL INDEX REGVAL
        SET PRINT ON
            ? CHR(27) + CHR(15)
              REPORT FORM VALRPT
            ? CHR(27) + CHR(18)
        SET PRINT OFF
      CLOSE DATABASES
      DO AGAIN
      LOOP
   ENDDO
************************
***STO/DISP/CAP RPT***
************************
  CASE CHOICE = 3
    @ 0,0 CLEAR
    DO PRINTER
    SELECT 1
      USE DISDIS INDEX REGDIS
    SELECT 2
      USE PKPLAN INDEX REGPARK
      SET RELATION TO REGNUM INTO DISDIS
    SELECT 3
      USE EXERCISE INDEX REGEXE
      SET RELATION TO REGNUM INTO PKPLAN
    SELECT 4
      USE MASTER INDEX REGMAST
      SET RELATION TO REGNUM INTO EXERCISE
    SELECT MASTER
       REPORT FORM SDC TO PRINT
       DO AGAIN
       LOOP
  ENDDO
*************************
***DISP/CHECKLIST RPT***
*************************
 CASE CHOICE = 4
  @ 0,0 CLEAR
  DO WHILE .T.
    DO PRINTER
    SELECT 1
      USE DISDIS INDEX REGDIS
    SELECT 2
```

```
      USE PKPLAN INDEX REGPARK
      SET RELATION TO REGNUM INTO DISDIS
    SELECT 3
      USE EXERCISE INDEX REGEXE
      SET RELATION TO REGNUM INTO PKPLAN
    SELECT 4
      USE MASTER INDEX REGMAST
      SET RELATION TO REGNUM INTO EXERCISE
    SELECT MASTER
       REPORT FORM DISPCHECK TO PRINT
       DO AGAIN
       LOOP
  ENDDO
***************
***SCHED RPT***
***************
CASE CHOICE = 5
 DO WHILE .T.
  STORE "RPT" TO NOUNS
  DO SEARCHER
    SELECT 1
      USE PKPLAN INDEX REGPARK
    SELECT 2
      USE EXERCISE INDEX REGEXE
      SET RELATION TO REGNUM INTO PKPLAN
    SELECT 3
      USE RELEASE INDEX REGREL
      SET RELATION TO REGNUM INTO EXERCISE
    SELECT 4
      USE MANMAINT INDEX REGMAN
      SET RELATION TO REGNUM INTO RELEASE
    SELECT 5
      USE VIMS INDEX REGVIMS
      SET RELATION TO REGNUM INTO MANMAINT
    SELECT 6
      USE MASTER INDEX REGMAST
      SET RELATION TO REGNUM INTO VIMS
    SELECT MASTER
    GO TOP
      READ
      FIND &REGNUM
         STORE MASTER->MGTCODE  TO MMGTCODE
         STORE MASTER->SPEQCODE TO MSPEQCODE
         STORE MASTER->MOBLCODE TO MMOBLCODE
         STORE MASTER->ASC      TO MASC
         STORE MASTER->DATEASGN TO MDATEASGN
         STORE VIMS->VEHTYPE    TO MVEHTYPE
         STORE VIMS->DATEACPT   TO MDATEACPT
         STORE VIMS->DATEWARX   TO MDATEWARX
         STORE VIMS->DTMX1DUE   TO MDTMX1DUE
         STORE VIMS->DTMX2DUE   TO MDTMX2DUE
         STORE VIMS->DTMX3DUE   TO MDTMX3DUE
         STORE VIMS->DTASIDUE   TO MDTASIDUE
         STORE VIMS->DATELOFDUE TO MDTLOFDUE
```

```
          STORE VIMS->TOTMILEAGE TO MTOTMILAGE
          STORE VIMS->MILE1DUE   TO MMILE1DUE
          STORE VIMS->MILE2DUE   TO MMILE2DUE
          STORE VIMS->MILE3DUE   TO MMILE3DUE
          STORE VIMS->MILEASI    TO MMILEASI
          STORE VIMS->MILELOF    TO MMILELOF
          STORE VIMS->MILEWARX   TO MMILEWARX
          STORE MANMAINT->MXSTATUS   TO MMXTATUS
          STORE MANMAINT->MXRTNDATE  TO MMXRTNDATE
          STORE MANMAINT->MXRMKS     TO MMXRMKS
          STORE MANMAINT->SIINTERVAL TO MSIINTERVAL
          STORE MANMAINT->SISWITCH   TO MSISWITCH
          STORE MANMAINT->LOFINTERV  TO MLOFINTER
          STORE MANMAINT->LOFSWITCH  TO MLOFSWITCH
          STORE EXERCISE->EXLSTDT TO MEXLSTDT
          STORE EXERCISE->EXNXTDT TO MEXNXTDT
          STORE EXERCISE->EXRMKS  TO MEXRMKS
          STORE RELEASE->RESTAT   TO MRESTAT
          STORE RELEASE->RERQDATE TO MRERQDATE
          STORE RELEASE->REACTDT  TO MREACTDT
          STORE RELEASE->REESTRTN TO MREESTRTN
          STORE RELEASE->RECASEN  TO MRECASEN
          STORE PKPLAN->PKLOTNO  TO MPKLOTNO
          STORE PKPLAN->PKROWNO  TO MPKROWNO
          STORE PKPLAN->PKCOLNO  TO MPKCOLNO
          STORE PKPLAN->PKSTATUS TO MPKSTATUS
SET DEVICE TO PRINT
@ 0,0 SAY "SCHEDULED ACTIONS REPORT FOR VEHICLE " +REGNUM
@ 0,50 SAY "PK LOT# "+ LTRIM(STR(MPKLOTNO))
@ 0,62 SAY "ROW " + LTRIM(STR(MPKROWNO))
@ 0,70 SAY "COL " + LTRIM(STR(MPKCOLNO))
@ 2,1  SAY "TOTAL VEHICLE LIFETIME MILEAGE " +
           LTRIM(STR(MTOTMILAGE)) + "     VEHICLE MANAGEMENT CODE " +
           MMGTCODE
@ 3,1  SAY "    DATE ACCEPTED BY AIR FORCE " + DTOC(MDATEACPT)
           +        "   SPECIAL EQUIPMENT CODE " + MSPEQCODE
@ 4,1  SAY "    DATE ASSIGNED TO WRM FLEET " + DTOC(MDATEASGN)
           +        "    ALLOWANCE SOURCE CODE " + MASC
@ 5,1  SAY " WRM IDENTIFIER CODE FROM VIMS " + MMOBLCODE  +
           "                      VEHICLE TYPE " + MVEHTYPE
@ 7,8 SAY "RELEASE CASE INFORMATION
          MAINTENANCE INFORMATION"
@ 9,1 SAY "        CURRENT RELEASE STATUS "  + MRESTAT +
          "                                  DATE DUE MILEAGE"
@ 10,1 SAY " CURRENT/LAST RELEASE CASE NO. " + MRECASEN +
          "     SCHEDULED MAINT. #1 " + DTOC(MDTMX1DUE) + " " +
          LTRIM(STR(MMILE1DUE))
@ 11,1 SAY "  DATE VEHICLE RELEASE DESIRED " + DTOC(MRERQDATE)
          +        "    SCHEDULED MAINT. #2 " + DTOC(MDTMX2DUE) + " " +
          LTRIM(STR(MMILE2DUE))
@ 12,1 SAY "DATE VEHICLE ACTUALLY RELEASED " + DTOC(MREACTDT)
          +        "    SCHEDULED MAINT. #3 " + DTOC(MDTMX3DUE) + " " +
          LTRIM(STR(MMILE3DUE))
@ 13,1 SAY "  ESTIMATED/ACTUAL RETURN DATE " + DTOC(MREESTRTN)
```

```
          +        "    ANNUAL SAFETY INSP. " + DTOC(MDTASIDUE) + " " +
          LTRIM(STR(MMILEASI))
@ 14,42 SAY "  SCHEDULED ON MONTHS IF YES " + MSISWITCH + "
          #MON " + LTRIM(STR(MSIINTERVAL))
@ 15,10 SAY "EXERCISE INFORMATION"  +                        "
          LUBE & OIL & FILTER " + DTOC(MDTLOFDUE) + " " +
          LTRIM(STR(MMILELOF))
@ 16,42 SAY "  SCHEDULED ON MONTHS IF YES " + MLOFSWITCH + "
          #MON " + LTRIM(STR(MLOFINTER))
@ 17,1  SAY "   DATE VEHICLE LAST EXERCISED " + DTOC(MEXLSTDT)
          +       "  CURRENT MAINT. STATUS " + MMXSTATUS
@ 18,1  SAY " DATE VEHICLE DUE FOR EXERCISE " + DTOC(MEXNXTDT)
          +       "          DATE DUE BACK " + DTOC(MMXRTNDATE)
@ 20,30 SAY "MAINTENANCE REMARKS"
@ 21,15 SAY MMXRMKS
@ 23,32 SAY "EXERCISE REMARKS"
@ 24,15 SAY MEXRMKS
@ 25,0  SAY
"
"-------------------------------------------------------------------------------
"
         EJECT
         SET DEVICE TO SCREEN
           CLOSE DATABASES
           DO AGAIN
           LOOP
  ENDDO
**************************
***SCHED CHECHKLIST RPT***
**************************
  CASE CHOICE = 6
   DO WHILE
       @ 0,0 CLEAR
       REPORT FORM SCHEDCK TO PRINT
       DO AGAIN
       LOOP
   ENDDO
************************
***REL CASE STAT RPT***
************************
  CASE CHOICE = 7
    @ 0,0 CLEAR
    @ 12,16 SAY "ENSURE THAT PRINTER IS ON AND PROPERLY
                 ALLIGNED"
    @ 23,24 SAY "****PRESS ANY KEY WHEN READY****"
       WAIT ""
    @ 0,0 CLEAR
    @ 12,27 SAY "PRINTING PLEASE BE PATIENT"
    CLEAR GETS
    SELECT 1
     USE PKPLAN INDEX REGPARK
    SELECT 2
     USE MASTER INDEX REGMAST
     SET RELATION TO REGNUM INTO PKPLAN
    SELECT 3
```

```
     USE RELEASE INDEX REGREL
     SET RELATION TO REGNUM INTO MASTER
    SELECT RELEASE
     GO TOP
     STORE 1 TO PAGENUMBER
     STORE 0 TO LINENUMBER
      SET PRINT ON
? "                              CURRENT RELEASE CASE STATUS"
? "                                 REPORT AS OF " DATE()
?
  DO WHILE .NOT. EOF() .AND. RESTAT = "O"
   IF LINENUMBER > 50 .OR. LINENUMBER = 50
      ?
      ?
      ?
      ?
      ?
      ? "                                              PAGE " +
        LTRIM(STR(PAGENUMBER))
      ?
      ?
      ?
      ?
      ?
      ?
      ?
      ?
      ?
      ?
? "                           CURRENT RELEASE CASE STATUS"
? "                              REPORT AS OF " DATE()
?
      STORE PAGENUMBER + 1 TO PAGENUMBER
      STORE 0 TO LINENUMBER
   ENDIF
         STORE RELEASE->REGNUM  TO MREGNUM
         STORE RELEASE->RESTAT  TO MRESTAT
         STORE RELEASE->REAPLVL TO MREAPLVL
         STORE RELEASE->REORG   TO MREORG
         STORE RELEASE->RENAMEPH TO MRENAMEPH
         STORE RELEASE->RERQDATE TO MRERQDATE
         STORE RELEASE->REACTDT  TO MREACTDT
         STORE RELEASE->REESTRTN TO MREESTRTN
         STORE RELEASE->RECASEN  TO MRECASEN
         STORE RELEASE->RECATEG  TO MRECATEG
         STORE RELEASE->RERMKS   TO MRERMKS
         STORE MASTER->MGTCODE  TO MMGTCODE
         STORE MASTER->NOMEN    TO MNOMEN
         STORE MASTER->ORGCODE  TO MORGCODE
         STORE MASTER->OWNCMD   TO MOWNCMD
         STORE MASTER->USECMD   TO MUSECMD
         STORE MASTER->SPEQCODE TO MSPEQCODE
         STORE MASTER->MOBLCODE TO MMOBLCODE
         STORE MASTER->ASC      TO MASC
```

```
         STORE PKPLAN->PKLOTNO  TO MPKLOTNO
         STORE PKPLAN->PKROWNO  TO MPKROWNO
         STORE PKPLAN->PKCOLNO  TO MPKCOLNO
         STORE PKPLAN->PKSTATUS TO MPKSTATUS
TEXT
```

```
                        VEHICLE IDENTIFICATION INFORMATION

         MGT                                    ORG  OWN USE
MOBL
 REGNUM  CODE         NOMENCLATURE              CODE CMD CMD
SPEQCODE  CODE   ASC
ENDTEXT
? MREGNUM + " " + MMGTCODE + " " + MNOMEN + "    " + MORGCODE +
"    " + MOWNCMD + "  " + MUSECODE + " " MSPEQCODE + "  " +
MMOBLCODE + "    " + MASC
?
TEXT

                                RELEASE CASE INFORMATION

 CASE                                              USER      RQD
ACCPT      EST
NUMBER          USER NAME, PHONE NUMBER            ORG      DATE
DATE      RTN
ENDTEXT
? MRECASEN + "        " + MRENAMEPH + "      " + MREORG + "     "
+ DTOC(MRERQDATE) + "  " + DTOC(MREACTDT) + " " +
DTOC(MREESTRTN)
?
? "        REMARKS: " + MRERMKS
?
TEXT
                                    PARKING INFORMATION

                  LOT NUMBER    ROW NUMBER  COLUMN NUMBER  PARK
STATUS
ENDTEXT
? "              " + STR(MPKLOTNO) + "    " + STR(MPKROWNO) + "
" + STR(MPKCOLNO) + "               " + MPKSTATUS
?
  STORE LINENUMBER + 25 TO LINENUMBER
      SKIP
ENDDO
  @ 0,0 CLEAR
  DO MAIN
******************
***SUMMARY RPT***
******************
  CASE CHOICE = 8
    DO WHILE .T.
       @ 0,0 CLEAR
         USE SUMMARY INDEX REGSUM
```

```
        REPORT FORM SUMRPT TO PRINT
        DO AGAIN
        LOOP
    ENDDO
  CASE CHOICE = 0
     @ 0,0 CLEAR
     CLOSE DATABASES
     DO MAIN
  ENDCASE

                              ****************************
                              *         PROCA.PRG        *
                              *standardized proced. file*
                              ****************************

*********************************
PROGAMMER: ROBERT S THOMAS 1988
    CALLS: MAIN.PRG
*********************************
*****PROC #1******
PROCEDURE SEARCHER
 @ 0,0 CLEAR
   CLEAR GETS
   STORE SPACE(8) TO REGNUM
 @ 11,16 TO 13,63 DOUBLE
 @ 12,17 SAY "ENTER THE VEHICLE REGISTRATION NUMBER "
 @ 12,55 GET REGNUM PICTURE "@! 99A99999"
  READ
  DO SWITCHER
  GO TOP
  FIND &REGNUM
     IF .NOT. FOUND()
         @ 0,0 CLEAR
         @ 7,12 TO 10,69 DOUBLE
         @ 8,14 SAY "***THIS REGISTRATION NUMBER IS NOT IN THE
                    DATABASE***"
         @ 9,14 SAY "DO YOU WISH TO ENTER ANOTHER REGISTRATION
                    NUMBER ? "
          STORE "Y" TO CHOICE
          @ 9,65 GET CHOICE PICTURE "Y"
            READ
           IF CHOICE = "Y"
              CLEAR GETS
              CLOSE DATABASES
              DO SEARCHER
           ENDIF
              CLOSE DATABASES
              CLEAR GETS
              DO MAIN
     ENDIF
RETURN
******PROC #2******
PROCEDUI E SWITCHER
     IF SWITCH = 1
```

```
         USE MASTER INDEX REGMAST
      ENDIF
      IF SWITCH = 2
         USE VIMS INDEX REGVIMS
      ENDIF
      IF SWITCH = 3
         USE VAL INDEX REGVAL
      ENDIF
      IF SWITCH = 4
         USE RELEASE INDEX REGREL
      ENDIF
      IF SWITCH = 5
         USE MANMAINT INDEX REGMAN
      ENDIF
      IF SWITCH = 6
         USE DISDIS INDEX REGDIS
      ENDIF
      IF SWITCH = 7
         USE PKPLAN INDEX REGPARK
      ENDIF
      IF SWITCH = 8
         USE EXERCISE INDEX REGEXE
      ENDIF
******PROC #3******
PROCEDURE SHOWSWITCH
  IF SWITCH = 1
     DO STOREMAST
     DO SHOWMAST
  ENDIF
  IF SWITCH = 2
     DO STOREVIMS
     DO SHOWVIMS
  ENDIF
  IF SWITCH = 3
     DO STOREVAL
     DO SHOWVAL
  ENDIF
  IF SWITCH = 4
     DO STOREREL
     DO SHOWREL
  ENDIF
  IF SWITCH = 5
     DO STOREMAN
     DO SHOWMAN
  ENDIF
  IF SWITCH = 6
     DO STOREDIS
     DO SHOWDIS
  ENDIF
  IF SWITCH = 7
     DO STOREPARK
     DO SHOWPARK
  ENDIF
  IF SWITCH = 8
```

```
      DO STOREEXE
      DO SHOWEXE
   ENDIF
   RETURN
******PROC #4******
PROCEDURE ADDER
    PARAMETER MREGNUM
    SET EXACT ON
    GO TOP
    LOCATE FOR REGNUM = MREGNUM
      IF EOF()
         STORE "N" TO OWRITE
         SET EXACT OFF
         RETURN
      ENDIF
      IF FOUND()
         @ 21,14 SAY "      THIS VEHICLE IS ALREADY IN THE
                     DATABASE !    "
         @ 22,14 SAY "      DO YOU WANT TO OVERWRITE THE RECORD
                     Y/N ?    "
           STORE "Y" TO CHOICE
         @ 22,61 GET CHOICE PICTURE "@! Y"
           READ
           IF CHOICE = "N"
             CLEAR GETS
             CLOSE DATABASES
             SET EXACT OFF
             DO MAIN
           ENDIF
             STORE "Y" TO OWRITE
             SET EXACT OFF
             RETURN
      ENDIF
 RETURN
******PROC #5******
PROCEDURE DELETER
@ 22,9 SAY "      IS THIS THE VEHICLE RECORD YOU WISH TO DELETE
           Y/N ?    "
   STORE "Y" TO ANSWER
   @ 22,70 GET ANSWER PICTURE "@! Y"
     READ
       IF ANSWER = "N"
          CLOSE DATABASE
          RETURN
       ENDIF
@ 12,1  SAY "DELETING RECORD # " + TRIM(RECNO()) + " FOR
            VEHICLE " + TRIM(REGNUM) + " FROM THE DATABASE"
          SET SAFETY ON
          DELETE FOR REGNUM = MREGNUM
          PACK
          CLOSE DATABASE
          SET SAFETY OFF
@ 0,0 CLEAR
@ 12,1  SAY "COPYING RECORD # " + TRIM(RECNO()) + " FOR
```

```
      VEHICLE " + TRIM(REGNUM) + " TO THE TEMPORARY DATABASE"
   IF SWITCH = 1
      USE TEMPMAST INDEX TMAST
      APPEND BLANK
        DO REPMAST
         CLEAR GETS
   ENDIF
   IF SWITCH = 2
      USE TEMPVAL INDEX TVAL
      APPEND BLANK
         DO REPVAL
         CLEAR GETS
   ENDIF
   IF SWITCH = 3
      USE TEMPVIMS INDEX TVIMS
      APPEND BLANK
         DO REPVIMS
         CLEAR GETS
   ENDIF
   IF SWITCH = 4
      USE TRELEASE INDEX TREL
      APPEND BLANK
         DO REPREL
         CLEAR GETS
   ENDIF
   IF SWITCH = 6
      USE TDISDIS INDEX TDIS
      APPEND BLANK
         DO REPDIS
         CLEAR GETS
   ENDIF
   IF SWITCH = 5
      USE TMANMAINT INDEX TMAN
      APPEND BLANK
         DO REPMAN
         CLEAR GETS
   ENDIF
   IF SWITCH = 7
      USE TPKPLAN INDEX TPK
      APPEND BLANK
         DO REPPARK
         CLEAR GETS
   ENDIF
   IF SWITCH = 8
      USE TEXERCISE INDEX TEXE
      APPEND BLANK
         DO REPEXE
         CLEAR GETS
   ENDIF
   CLOSE DATABASES
RETURN
******PROC #6******
PROCEDURE UNDELETER
● 21,25 SAY "IS THIS THE VEHICLE RECORD YOU WISH TO UNDELETE ? "
```

```
STORE "Y" TO ANSWER
@ 21,66 GET ANSWER PICTURE "@! Y"
  IF ANSWER = "N"
     RETURN
  ENDIF
   STORE MREGNUM TO REGNUM
     IF SWITCH = 1
        USE MASTER INDEX REGMAST
         DO FINDER
          APPEND BLANK
          DO REPMAST
          CLEAR GETS
     ENDIF
     IF SWITCH = 2
        USE VAL INDEX REGVAL
         DO FINDER
          APPEND BLANK
          DO REPVAL
          CLEAR GETS
     ENDIF
     IF SWITCH = 3
        USE VIMS INDEX REGVIMS
         DO FINDER
          APPEND BLANK
          DO REPVIMS
          CLEAR GETS
     ENDIF
     IF SWITCH = 4
        USE RELEASE INDEX REGREL
         DO FINDER
          APPEND BLANK
          DO REPREL
          CLEAR GETS
     ENDIF
     IF SWITCH = 5
        USE MANMAINT INDEX REGMAN
         DO FINDER
          APPEND BLANK
          DO REPMAN
          CLEAR GETS
     ENDIF
     IF SWITCH = 6
        USE DISDIS INDEX REGDIS
         DO FINDER
          APPEND BLANK
          DO REPDIS
          CLEAR GETS
     ENDIF
     IF SWITCH = 7
        USE PKPLAN INDEX REGPARK
         DO FINDER
          APPEND BLANK
          DO REPPARK
          CLEAR GETS
```

```
        ENDIF
        IF SWITCH = 8
           USE EXERCISE INDEX REGEXE
            DO FINDER
             APPEND BLANK
             DO REPEXE
             CLEAR GETS
        ENDIF
CLEAR GETS
RETURN
******PROC #7******
PROCEDURE SHOWREL
@  0,0 CLEAR
@  1,5 TO 23,75 DOUBLE
@  4,6 TO 4,74 DOUBLE
@  2,25 SAY MTITLE
@  3,20 SAY RTRIM(NOUNS) + " A VEHICLE (RELEASE DATABASE)"
@  5,7  SAY "          VEHICLE REGISTRATION NUMBER"
        @ 5,48 SAY MREGNUM
@  6,7  SAY "                CURRENT RELEASE STATUS"
        @ 6,48 SAY MRESTAT
@  7,7  SAY "                CURRENT APPROVAL LEVEL"
        @ 7,48 SAY MREAPLVL
@  8,7  SAY "ORGANIZATION CODE VEHICLE RELEASED TO"
        @ 8,48 SAY MREORG
@  9,7  SAY "NAME AND PHONE NUMBER OF VEHICLE USER"
        @ 9,48 SAY MRENAMEPH
@ 10,7  SAY "         DATE VEHICLE RELEASE DESIRED"
        @ 10,48 SAY DTOC(MRERQDATE)
@ 11,7  SAY "       DATE VEHICLE ACTUALLY RELEASED"
        @ 11,48 SAY DTOC(MREACTDT)
@ 12,7  SAY "         ESTIMATED/ACTUAL RETURN DATE"
        @ 12,48 SAY DTOC(MREESTRTN)
@ 13,7  SAY "        CURRENT/LAST RELEASE CASE NO."
        @ 13,48 SAY MRECASEN
@ 14,7  SAY "   CURRENT/LAST RELEASE CASE CATEGORY"
        @ 14,48 SAY MRECATEG
@ 16,6 TO 16,74 DOUBLE
@ 17,7  SAY "                                    REMARKS"
        @ 18,15 SAY MRERMKS
@ 20,6  TO 20,74 DOUBLE
@ 21,23 SAY "****PRESS ANY KEY TO CONTINUE****"
        WAIT ""
RETURN
******PROC #8******
PROCEDURE GETREL
@  0,0 CLEAR
@  1,5 TO 23,75 DOUBLE
@  4,6 TO 4,74 DOUBLE
@  2,25 SAY MTITLE
@  3,20 SAY RTRIM(NOUNS) + " A VEHICLE (RELEASE DATABASE)"
@  5,7  SAY "          VEHICLE REGISTRATION NUMBER"
        @ 5,48 GET MREGNUM PICTURE "@! 99A99999"
@  6,7  SAY "                CURRENT RELEASE STATUS"
```

```
        @ 6,48 GET MRESTAT PICTURE "@! X"
@  7,7  SAY "                    CURRENT APPROVAL LEVEL"
        @ 7,48 GET MREAPLVL PICTURE "@! X"
@  8,7  SAY "ORGANIZATION CODE VEHICLE RELEASED TO"
        @ 8,48 GET MREORG PICTURE "@! XX"
@  9,7  SAY "NAME AND PHONE NUMBER OF VEHICLE USER"
        @ 9,48 GET MRENAMEPH PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXXX"
@ 10,7  SAY "              DATE VEHICLE RELEASE DESIRED"
        @ 10,48 GET MRERQDATE PICTURE "99/99/99"
@ 11,7  SAY "          DATE VEHICLE ACTUALLY RELEASED"
        @ 11,48 GET MREACTDT PICTURE "99/99/99"
@ 12,7  SAY "              ESTIMATED/ACTUAL RETURN DATE"
        @ 12,48 GET MREESTRTN PICTURE "99/99/99"
@ 13,7  SAY "            CURRENT/LAST RELEASE CASE NO."
        @ 13,48 GET MRECASEN PICTURE "99-9999"
@ 14,7  SAY "   CURRENT/LAST RELEASE CASE CATEGORY"
        @ 14,48 GET MRECATEG PICTURE "@! X"
@ 17,7  SAY "                                        REMARKS"
@ 18,15 GET MRERMKS PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 20,6 TO 20,74 DOUBLE
@ 21,18 SAY "***ENTER DATA IN BLOCKED AREAS PRESS RETURN***"
@ 22,20 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO EDIT***"
        READ
           STORE 4 TO SWITCH
              DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPREL
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPREL
         CLOSE DATABASES
  RETURN
******PROC # 9 ******
PROCEDURE REPREL
REPLACE REGNUM WITH MREGNUM
REPLACE RESTAT WITH MRESTAT
REPLACE REAPLVL WITH MREAPLVL
REPLACE REORG WITH MREORG
REPLACE RENAMEPH WITH MRENAMEPH
REPLACE RERQDATE WITH CTOD(MRERQDATE)
REPLACE REACTDT WITH  CTOD(MREACTDT)
REPLACE REESTRTN WITH CTOD(MREESTRTN)
REPLACE RECASEN WITH MRECASEN
REPLACE RECATEG WITH MRECATEG
REPLACE RERMKS WITH MRERMKS
CLEAR GETS
RETURN
******PROC #10******
PROCEDURE REPEATER
@ 21,14 SAY "
```

```
"
@ 22,14 SAY "DO YOU WISH TO " + TRIM(NOUNS) + " ANOTHER
VEHICLE RECORD ?    "
     STORE 'Y' TO CHOICE
@ 22,69 GET CHOICE PICTURE "@! Y"
  READ
  IF CHOICE = "N"
    CLEAR GETS
    CLEAR
    DO MAIN
  ENDIF
    CLOSE DATABASES
    CLEAR GETS
  RETURN
******PROC #11******
PROCEDURE STOREREL
STORE REGNUM TO MREGNUM
STORE RESTAT TO MRESTAT
STORE REAPLVL TO MREAPLVL
STORE REORG TO MREORG
STORE RENAMEPH TO MRENAMEPH
STORE RERQDATE TO MRERQDATE
STORE REACTDT TO MREACTDT
STORE REESTRTN TO MREESTRTN
STORE RECASEN TO MRECASEN
STORE RECATEG TO MRECATEG
STORE RERMKS TO MRERMKS
RETURN
******PROC #12******
PROCEDURE SHOWEXE
@  0,0 CLEAR
@  3,13 TO 20,67 DOUBLE
@  6,14 TO 6,66 DOUBLE
@  4,25 SAY MTITLE
@  5,20 SAY RTRIM(NOUNS) + " A VEHICLE (EXERCISE DATABASE)"
@  8,20 SAY "  VEHICLE REGISTRATION NUMBER "
   @ 8,50 SAY MREGNUM
@ 10,20 SAY "  DATE VEHICLE LAST EXERCISED "
   @ 10,50 SAY DTOC(MEXLSTDT)
@ 12,20 SAY "DATE VEHICLE DUE FOR EXERCISE "
   @ 12,50 SAY DTOC(MEXNXTDT)
@ 13,14 TO 13,66 DOUBLE
@ 14,36 SAY "REMARKS"
   @ 15,15 SAY MEXRMKS
@ 16,14 TO 16,66 DOUBLE
@ 18,23 SAY "****PRESS ANY KEY TO CONTINUE****"
   WAIT ""
RETURN
******PROC #13******
PROCEDURE GETEXE
@   0,0 CLEAR
@   3,13 TO 20,67 DOUBLE
@   6,14 TO 6,66 DOUBLE
@   4,25 SAY MTITLE
```

```
@   5,20 SAY RTRIM(NOUNS) + " A VEHICLE (EXERCISE DATABASE)"
@   8,15 SAY "  VEHICLE REGISTRATION NUMBER "
        @ 8,45 GET MREGNUM PICTURE "@! 99A99999"
@  10,15 SAY "  DATE VEHICLE LAST EXERCISED "
        @ 10,45 GET MEXLSTDT PICTURE "99/99/99"
@  12,15 SAY "DATE VEHICLE DUE FOR EXERCISE "
        @ 12,45 GET MEXNXTDT PICTURE "99/99/99"
@  13,14 TO 13,66 DOUBLE
@  14,36 SAY "REMARKS"
@  15,15 GET MEXRMKS PICTURE
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@  16,14 TO 16,66 DOUBLE
@  21,18 SAY "***ENTER DATA IN BLOCKED AREAS PRESS RETURN***"
@  22,20 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO EDIT***"
      READ
         STORE 8 TO SWITCH
            DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPEXE
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPEXE
         CLOSE DATABASES
  RETURN
******PROC #14******
PROCEDURE REPEXE
REPLACE REGNUM WITH MREGNUM
REPLACE EXLSTDT WITH MEXLSTDT
REPLACE EXNXTDT WITH MEXNXTDT
REPLACE EXRMKS  WITH MEXRMKS
CLEAR GETS
RETURN
******PROC #15******
PROCEDURE STOREEXE
STORE REGNUM TO MREGNUM
STORE EXLSTDT TO MEXLSTDT
STORE EXNXTDT TO MEXNXTDT
STORE EXRMKS TO MEXRMKS
RETURN
******PROC #16******
PROCEDURE SHOWMAN
@  0,0 CLEAR
@  0,1 TO 24,79 DOUBLE
@  5,2 TO 5,78 DOUBLE
@  2,25 SAY MTITLE
@  4,20 SAY RTRIM(NOUNS) + " A VEHICLE (MANMAINT DATABASE)"
@  7,7 SAY "   VEHICLE REGISTRATION NUMBER"
      @ 7,48 SAY MREGNUM
@  8,7 SAY "    CURRENT MAINTENANCE STATUS"
      @ 8,48 SAY MMXSTATUS
@  9,7 SAY " DATE VEHICLE DUE BACK FROM MX"
      @ 9,48 SAY DTOC(MMXRTNDATE)
```

```
@ 10,7 SAY " MONTHS BETWEEN SCHEDULED SI'S"
       @ 10,48 SAY MSIINTERVAL
@ 11,7 SAY " SCHEDULE SI'S BASED ON MONTHS"
       @ 11,48 SAY MSISWITCH
@ 12,7 SAY "MONTHS BETWEEN SCHEDULED LOF'S"
       @ 12,48 SAY MLOFINTER
@ 13,7 SAY "SCHEDULE LOF'S BASED ON MONTHS"
       @ 13,48 SAY MLOFSWITCH
@ 15,2 TO 15,78 DOUBLE
@ 16,7 SAY "                          REMARKS"
       @ 18,15 SAY MMXRMKS
@ 20,2 TO 20,78 DOUBLE
@ 21,23 SAY "****PRESS ANY KEY TO CONTINUE****"
       WAIT ""
RETURN
******PROC #17******
PROCEDURE GETMAN
@  0,0 CLEAR
@  0,1 TO 24,79 DOUBLE
@  5,6 TO 5,78 DOUBLE
@  2,25 SAY MTITLE
@  4,20 SAY RTRIM(NOUNS) + " A VEHICLE (MANMAINT DATABASE)"
@  7,7 SAY "   VEHICLE REGISTRATION NUMBER"
       @ 7,48 GET MREGNUM PICTURE "@! 99A99999"
@  8,7 SAY "    CURRENT MAINTENANCE STATUS"
       @ 8,48 GET MMXSTATUS PICTURE "@! X"
@  9,7 SAY " DATE VEHICLE DUE BACK FROM MX"
       @ 9,48 GET MMXRTNDATE PICTURE "99/99/99"
@ 10,7 SAY " MONTHS BETWEEN SCHEDULED SI'S"
       @ 10,48 GET MSIINTERVAL PICTURE "99"
@ 11,7 SAY " SCHEDULE SI'S BASED ON MONTHS"
       @ 11,48 GET MSISWITCH PICTURE "@! Y"
@ 12,7 SAY "MONTHS BETWEEN SCHEDULED LOF'S"
       @ 12,48 GET MLOFINTER PICTURE "99"
@ 13,7 SAY "SCHEDULE LOF'S BASED ON MONTHS"
       @ 13,48 GET MLOFSWITCH PICTURE "@! Y"
@ 15,2 TO 15,78 DOUBLE
@ 16,7 SAY "                                    REMARKS"
@ 18,15 GET MMXRMKS PICTURE
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 20,2 TO 20,78 DOUBLE
@ 21,18 SAY "***ENTER DATA IN BLOCKED AREAS PRESS RETURN***"
@ 22,20 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO EDIT***"
       READ
          STORE 5 TO SWITCH
             DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPMAN
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPMAN
         CLOSE DATABASES
```

```
  RETURN
******PROC #17******
PROCEDURE REPMAN
REPLACE REGNUM WITH MREGNUM
REPLACE MXSTATUS WITH MMXSTATUS
REPLACE MXRTNDATE WITH CTOD(MMXRTNDATE)
REPLACE SIINTERVAL WITH MSIINTERVAL
REPLACE SISWITCH WITH MSISWITCH
REPLACE LOFINTERV WITH MLOFINTER
REPLACE LOFSWITCH WITH MLOFSWITCH
REPLACE MXRMKS WITH MMXRMKS
CLEAR GETS
RETURN
******PROC #18******
PROCEDURE STOREMAN
STORE REGNUM TO MREGNUM
STORE MXSTATUS TO MMXSTATUS
STORE MXRTNDATE TO MMXRTNDATE
STORE SIINTERVAL TO MSIINTERVAL
STORE SISWITCH TO MSISWITCH
STORE LOFINTERV TO MLOFINTER
STORE LOFSWITCH TO MLOFSWITCH
STORE MXRMKS TO MMXRMKS
RETURN
******PROC #19******
PROCEDURE SHOWPARK
 @ 0,0 clear
 @ 7,11 to 23,67 double
 @ 10,12 to 10,66 double
 @ 20,12 TO 20,66 DOUBLE
 @ 8,25 say MTITLE
 @ 9,20 say RTRIM(NOUNS) + " A VEHICLE (PKPLAN DATABASE)"
     @ 12,23 say "Vehicle Registration Number"
             @ 12,54 say Mregnum
     @ 13,16 say "Vehicle Storage Parking Lot Number"
             @ 13,52 say Mpklotno
     @ 14,20 say "Vehicle Parking Lot Row Number"
             @ 14,52 say Mpkrowno
     @ 15,17 say "Vehicle Parking Lot Column Number"
             @ 15,52 say Mpkcolno
     @ 16,28 say "Vehicle Current Status"
             @ 16,61 say Mpkstatus
RETURN
******PROC #20******
PROCEDURE GETPARK
@   0,0 CLEAR
@   7,15 to 23,65 double
@  10,16 to 10,64 double
@  18,16 TO 18,64 DOUBLE
@ 8,25 say MTITLE
@ 9,20 say RTRIM(NOUNS) + " A VEHICLE (PKPLAN DATABASE)"
       @ 12,25 SAY "VEHICLE REGISTRATION NUMBER"
               @ 12,52 GET MREGNUM PICTURE "@! 99A99999"
       @ 13,19 SAY "VEHICLE STORAGE PARKING LOT NUMBER"
```

```
              @ 13,52 GET MPKLOTNO PICTURE "999"
       @ 14,23 SAY "VEHICLE PARKING LOT ROW NUMBER"
              @ 14,52 GET MPKROWNO PICTURE "999"
       @ 15,20 SAY "VEHICLE PARKING LOT COLUMN NUMBER"
              @ 15,52 GET MPKCOLNO PICTURE "999"
       @ 16,30 SAY "VEHICLE CURRENT STATUS"
              @ 16,52 GET MPKSTATUS PICTURE "@! A"
       @ 19,18 SAY "***ENTER DATA IN BLOCKED AREAS PRESS
RETURN***"
       @ 20,20 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO
EDIT***"
         READ
            STORE 7 TO SWITCH
             DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPPARK
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPPARK
         CLOSE DATABASES
 RETURN
******PROC #21******
PROCEDURE REPPARK
REPLACE REGNUM WITH MREGNUM
REPLACE PKLOTNO WITH MPKLOTNO
REPLACE PKROWNO WITH MPKROWNO
REPLACE PKCOLNO WITH MPKCOLNO
REPLACE PKSTATUS WITH MPKSTATUS
CLEAR GETS
RETURN
******PROC #22******
PROCEDURE STOREPARK
STORE    REGNUM TO MREGNUM
STORE   PKLOTNO TO MPKLOTNO
STORE   PKROWNO TO MPKROWNO
STORE   PKCOLNO TO MPKCOLNO
STORE  PKSTATUS TO MPKSTATUS
RETURN
******PROC #23******
PROCEDURE SHOWDIS
@ 0,0 CLEAR
@  3,5 TO 24,75 DOUBLE
@  6,6 TO 6,74 DOUBLE
@  4,25 SAY MTITLE
@  5,20 SAY RTRIM(NOUNS) + " A VEHICLE (DISDIS DATABASE)"
@  8,10  SAY "      VEHICLE REGISTRATION NUMBER"
       @ 8,45  SAY MREGNUM
@  9,10  SAY "               DISPERSAL LOCATION"
       @ 9,45  SAY MDISPDEST
@ 10,10  SAY "DATE REQUIRED AT WARTIME LOCATION"
       @ 10,45  SAY MDISPRQDT
@ 11,10  SAY "    MANHOURS NEEDED TO DISTRIBUTE"
```

```
        @ 11,45 SAY MDISPMHRS
@ 13,6 TO 13,74 DOUBLE
@ 14,31 SAY "DISPERSAL REMARKS"
        @ 15,15 SAY MDISRMK1
@ 16,6 TO 16,75 DOUBLE
@ 17,25 SAY "SPECIAL DISPERSAL INSTRUCTIONS"
        @ 18,15 SAY MDISINST
@ 20,6 TO 20,75 DOUBLE
@ 22,23 SAY "****PRESS ANY KEY TO CONTINUE****"
  WAIT ""
RETURN
******PROC #24******
PROCEDURE GETDIS
@  0,0 CLEAR
@  3,5 TO 24,75 DOUBLE
@  6,6 TO  6,74 DOUBLE
@  4,25 SAY MTITLE
@  5,20 SAY RTRIM(NOUNS) + " A VEHICLE (DISDIS DATABASE)"
@  8,10 SAY "      VEHICLE REGISTRATION NUMBER"
       @ 8,45 GET MREGNUM PICTURE "@! 99A99999"
@  9,10 SAY "                DISPERSAL LOCATION"
       @ 9,45 GET MDISPDEST PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXX"
@ 10,10 SAY "DATE REQUIRED AT WARTIME LOCATION"
       @ 10,45 GET MDISPRQDT PICTURE "999"
@ 11,10 SAY "    MANHOURS NEEDED TO DISTRIBUTE"
       @ 11,45 GET MDISPMHRS PICTURE "999"
@ 13,6 TO 13,74 DOUBLE
@ 14,31 SAY "DISPERSAL REMARKS"
       @ 15,15 GET MDISRMK1 PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 16,6 TO 16,74 DOUBLE
@ 17,25 SAY "SPECIAL DISPERSAL INSTRUCTIONS"
       @ 18,15 GET MDISINST PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 20,6 TO 20,74 DOUBLE
@ 21,18 SAY "***ENTER DATA IN BLOCKED AREAS PRESS RETURN***"
@ 22,20 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO EDIT***"
 READ
      DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPDIS
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPDIS
         CLOSE DATABASES
 RETURN
******PROC #25******
PROCEDURE REPDIS
 REPLACE REGNUM   WITH MREGNUM
 REPLACE DISPDEST WITH MDISPDEST
 REPLACE DISPRQDT WITH MDISPRQDT
```

```
 REPLACE DISPMHRS WITH MDISPMHRS
 REPLACE DISRMK1  WITH MDISRMK1
 REPLACE DISINST  WITH MDISINST
   CLEAR GETS
RETURN
******PROC #26******
PROCEDURE STOREDIS
STORE REGNUM   TO MREGNUM
STORE DISPDEST TO MDISPDEST
STORE DISPRQDT TO MDISPRQDT
STORE DISPMHRS TO MDISPMHRS
STORE DISRMK1  TO MDISRMK1
STORE DISINST  TO MDISINST
RETURN
******PROC #27******
PROCEDURE GETMAST
@  0,0 CLEAR
@  0,0 to 24,79 double
@  5,1 to 5,78 double
@  20,1 to 20,78 double
@  2,25 SAY MTITLE
@  4,20 SAY RTRIM(NOUNS) + " A VEHICLE (MASTER DATABASE)"
@  7,1 SAY "REGISTRATION NUMBER"
       @ 7,22 GET MREGNUM PICTURE "@! 99A99999"
@  8,1 SAY "MANAGEMENT CODE"
       @  8,22 GET MMGTCODE PICTURE "@! A999"
@  9,1 SAY "ASSIGNED MGT CODE"
       @  9,22 GET MASGNMGT PICTURE "@! A999"
@ 10,1 SAY "MASTER NAT. STOCK NO."
       @ 10,22 GET MISNUM PICTURE "@! XXXXXXXXXXXXXXXX"
@ 11,1 SAY "VEHICLE NAT STOCK NO "
       @ 11,22 GET MNSN PICTURE "@! XXXXXXXXXXXXXXXXX"
@ 12,1 SAY "ASSIGNED NSN"
       @ 12,22 GET MASGNIS PICTURE "@! XXXXXXXXXXXXXXXXX"
@ 13,1 SAY "VEH. NOMENCLATURE"
       @ 13,22 GET MNOMEN PICTURE "@!
XXXXXXXXXXXXXXXXXXXXXXXX"
@ 14,1 SAY "ORGANIZATION CODE"
       @ 14,22 GET MORGCODE PICTURE "@! XX"
@ 15,1 SAY "OWNING MAJ. COMMAND"
       @ 15,22 GET MOWNCMD PICTURE "@! XX"
@ 16,1 SAY "USING MAJ. COMMAND"
       @ 16,22 GET MUSECMD PICTURE "@! XX"
@  7,47 SAY "SUIT./UNSUIT. SUB."
        @  7,69 GET MITEMCODE PICTURE "@! X"
@  8,47 SAY "VEH. REPLACEMENT CODE"
        @  8,69 GET MREPCODE PICTURE "@! X"
@  9,47 SAY "WRM USE CATEGORY"
        @  9,69 GET MUSECODE PICTURE "@! X"
@ 10,47 SAY "SPECIAL EQUIP. CODE"
       @ 10,69 GET MSPEQCODE PICTURE "@! XXXXXXXXX"
@ 11,47 SAY "DATE ASSGN. WRM FLT."
       @ 11,69 GET MDATEASGN PICTURE "99/99/99"
@ 12,47 SAY "WRM/VIMS IDENT. CODE"
```

```
        @ 12,69 GET MMOBLCODE PICTURE "@! X"
@ 13,47 SAY "DATE VEH. DUE IN"
        @ 13,69 GET MDATEDUE PICTURE "99/99/99"
@ 14,47 SAY "ALLOWANCE SOURCE CODE"
        @ 14,69 GET MASC PICTURE "@! XXXXXX"
@ 21,17 SAY "***ENTER DATA IN BLOCKED AREAS PRESS RETURN***"
@ 22,19 SAY "***CURSOR UP/DOWN OR RIGHT/LEFT TO EDIT***"
        read
           STORE 1 TO SWITCH
              DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPMAST
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPMAST
         CLOSE DATABASES
  RETURN
******PROC #28******
PROCEDURE REPMAST
REPLACE REGNUM WITH MREGNUM
REPLACE MGTCODE WITH MMGTCODE
REPLACE ASGNMGT WITH MASGNMGT
REPLACE ISNUM WITH MISNUM
REPLACE NSN WITH MNSN
REPLACE ASGNIS WITH MASGNIS
REPLACE NOMEN WITH MNOMEN
REPLACE ORGCODE WITH MORGCODE
REPLACE OWNCMD WITH MOWNCMD
REPLACE USECMD WITH MUSECMD
REPLACE ITEMCODE WITH MITEMCODE
REPLACE REPCODE WITH MREPCODE
REPLACE USECODE WITH MUSECODE
REPLACE SPEQCODE WITH MSPEQCODE
REPLACE DATEASGN WITH CTOD(MDATEASGN)
REPLACE MOBLCODE WITH MMOBLCODE
REPLACE DATEDUE WITH CTOD(MDATEDUE)
REPLACE ASC WITH MASC
CLEAR GETS
RETURN
******PROC #29******
PROCEDURE SHOWMAST
@  0,0 CLEAR
@  1,0 TO 24,79 DOUBLE
@  5,1 TO 5,78 DOUBLE
@ 20,1 TO 20,78 DOUBLE
@  2,25 SAY MTITLE
@  4,20 SAY RTRIM(NOUNS) + " A VEHICLE (MASTER DATABASE)"
@  7,1 SAY "REGISTRATION NUMBER"
       @ 7,22 SAY MREGNUM
@  8,1 SAY "MANAGEMENT CODE"
       @ 8,22 SAY MMGTCODE
@  9,1 SAY "ASSIGNED MGT CODE"
```

```
        @ 9,22 SAY MASGNMGT
@ 10,1 SAY "MASTER NAT. STOCK NO."
        @ 10,22 SAY MISNUM
@ 11,1 SAY "VEHICLE NAT. STOCK NO"
        @ 11,22 SAY MNSN
@ 12,1 SAY "ASSIGNED NSN"
        @ 12,22 SAY MASGNIS
@ 13,1 SAY "VEH. NOMENCLATURE"
        @ 13,22 SAY MNOMEN
@ 14,1 SAY "ORGANIZATION CODE"
        @ 14,22 SAY MORGCODE
@ 15,1 SAY "OWNING MAJ. COMMAND"
        @ 15,22 SAY MOWNCMD
@ 16,1 SAY "USING MAJ. COMMAND"
        @ 16,22 SAY MUSECODE
@  7,47 SAY "SUIT/UNSUIT. SUB."
        @ 7,69 SAY MITEMCODE
@  8,47 SAY "VEH. REPLACEMENT CODE"
        @ 8,69 SAY MREPCODE
@  9,47 SAY "WRM USE CATEGORY"
        @ 9,69 SAY MUSECODE
@ 10,47 SAY "SPECIAL EQUIP. CODE"
        @ 10,69 SAY MSPEQCODE
@ 11,47 SAY "DATE ASSGN. WRM FLT."
        @ 11,69 SAY DTOC(MDATEASGN)
@ 12,47 SAY "WRM/VIMS IDENT. CODE"
        @ 12,69 SAY MMOBLCODE
@ 13,47 SAY "DATE VEH. DUE IN"
        @ 13,69 SAY DTOC(MDATEDUE)
@ 14,47 SAY "ALLOWANCE SOURCE CODE"
        @ 14,69 SAY MASC
@ 21,23 SAY "****PRESS ANY KEY TO CONTINUE****"
       WAIT ""
RETURN
******PROC #30******
PROCEDURE STOREMAST
STORE REGNUM   TO MREGNUM
STORE MGTCODE  TO MMGTCODE
STORE ASGNMGT  TO MASGNMGT
STORE ISNUM    TO MISNUM
STORE NSN      TO MNSN
STORE ASGNIS   TO MASGNIS
STORE NOMEN    TO MNOMEN
STORE ORGCODE  TO MORGCODE
STORE OWNCMD   TO MOWNCMD
STORE USECMD   TO MUSECMD
STORE ITEMCODE TO MITEMCODE
STORE REPCODE  TO MREPCODE
STORE USECODE  TO MUSECODE
STORE SPEQCODE TO MSPEQCODE
STORE DATEASGN TO MDATEASGN
STORE MOBLCODE TO MMOBLCODE
STORE DATEDUE  TO MDATEDUE
STORE ASC      TO MASC
```

```
RETURN
******PROC #31******
PROCEDURE GETVIMS
@ 0,0 CLEAR
@ 0,1 to 24,79 double
@ 3,2 to 3,78 double
@ 20,2 to 20,78 double
@ 1,25 SAY MTITLE
@ 2,20 SAY RTRIM(NOUNS) + " A VEHICLE (VIMS DATABASE)"
@ 4,16 say "VEHICLE REGISTRATION NUMBER"
       @ 4,44 GET MREGNUM PICTURE "@! 99A99999"
@ 5,12 say " TOTAL VEHICLE LIFETIME MILEAGE"
       @ 5,44 GET MTOTMILAGE PICTURE "999999"
       @ 5,55 SAY "(<999,999 MILES)"
@ 6,14 SAY " VEHICLE MILE/KM/HR/UNIT TYPE"
       @ 6,44 GET MVEHTYPE PICTURE "@! A"
@ 7,13 SAY " DATE SCHEDULED MAINTENANCE #1"
       @ 7,44 GET MDTMX1DUE PICTURE "99/99/99"
       @ 7,55 SAY "(MM/DD/YY)"
@ 8,13 SAY " DATE SCHEDULED MAINTENANCE #2"
       @ 8,44 GET MDTMX2DUE PICTURE "99/99/99"
       @ 8,55 SAY "(MM/DD/YY)"
@ 9,13 SAY " DATE SCHEDULED MAINTENANCE #3"
       @ 9,44 GET MDTMX3DUE PICTURE "99/99/99"
       @ 9,55 SAY "(MM/DD/YY)"
@ 10,14 SAY " DATE ANNUAL INSPECTION SCHEDULED"
        @ 10,44 get Mdtasidue picture "99/99/99"
        @ 10,55 say "(MM/DD/YY)"
@ 11,14 say " Date Lube, Oil & Filter Due "
        @ 11,44 get Mdtlofdue picture "99/99/99"
        @ 11,55 say "(MM/DD/YY)"
@ 12,15 say "Date Vehicle Accepted by AF "
        @ 12,44 get Mdateacpt picture "99/99/99"
        @ 12,55 say "(MM/DD/YY)"
@ 13,13 say " Date Vehicle Warranty Expires"
        @ 13,44 get Mdatewarx picture "99/99/99"
        @ 13,55 say "(MM/DD/YY)"
@ 14,15 say " Mileage Maintenance #1 due "
        @ 14,44 get Mmile1due picture "999999"
        @ 14,55 say "(<999,999 miles)"
@ 15,15 say " Mileage Maintenance #2 due "
        @ 15,44 get Mmile2due picture "999999"
        @ 15,55 say "(<999,999 miles)"
@ 16,16 say " Mileage Maintenance #3 due"
        @ 16,44 get Mmile3due picture "999999"
        @ 16,55 say "(<999,999 miles)"
@ 17,16 say " Mileage Annual Safety Due "
        @ 17,44 get Mmileasi picture "999999"
        @ 17,55 say "(<999,999 miles)"
@ 18,18 say " Mileage Warranty Expires"
        @ 18,44 get Mmilewarx picture "999999"
        @ 18,55 say "(<999,999 miles)"
@ 19,12 say " Mileage Lube, Oil $ Filter due"
        @ 19,44 get Mmilelof picture "999999"
```

```
        @ 19,55 say "(<999,999 miles)"
@ 22,14 say "****Enter Data in Blocked Areas and Press
Return****"
@ 23,19 say "***Cursor Up/Down or Right/Left to Edit***"
        read
           STORE 2 TO SWITCH
              DO ADDER WITH MREGNUM
      IF OWRITE = "Y"
         DO REPVIMS
         CLOSE DATABASES
         RETURN
      ENDIF
         APPEND BLANK
         DO REPVIMS
         CLOSE DATABASES
  RETURN
******PROC #32******
PROCEDURE REPVIMS
REPLACE REGNUM WITH MREGNUM
REPLACE TOTMILEAGE WITH MTOTMILAGE
REPLACE VEHTYPE WITH MVEHTYPE
REPLACE DTMX1DUE WITH CTOD(MDTMX1DUE)
REPLACE DTMX2DUE WITH CTOD(MDTMX2DUE)
REPLACE DTMX3DUE WITH CTOD(MDTMX3DUE)
REPLACE DTASIDUE WITH CTOD(MDTASIDUE)
REPLACE DATELOFDUE WITH CTOD(MDTLOFDUE)
REPLACE DATEACPT WITH CTOD(MDATEACPT)
REPLACE DATEWARX WITH CTOD(MDATEWARX)
REPLACE MILE1DUE WITH MMILE1DUE
REPLACE MILE2DUE WITH MMILE2DUE
REPLACE MILE3DUE WITH MMILE3DUE
REPLACE MILEASI  WITH MMILEASI
REPLACE MILEWARX WITH MMILEWARX
REPLACE MILELOF  WITH MMILELOF
CLEAR GETS
RETURN
******PROC #33******
PROCEDURE SHOWVIMS
@ 0,0 CLEAR
@ 0,1 TO 24,79 DOUBLE
@ 3,2 TO 3,78 DOUBLE
@ 20,2 TO 20,78 DOUBLE
@ 1,25 SAY MTITLE
@ 2,20 SAY RTRIM(NOUNS) + " A VEHICLE (VIMS DATABASE)"
@ 4,20 say "Vehicle Registration Number"
       @ 4,51 say Mregnum
@ 5,16 say " Total Vehicle Lifetime Mileage"
       @ 5,47 say Mtotmilage
@ 6,18 say " Vehicle Mile/Km/Hr/Unit Type"
       @ 6,51 say Mvehtype
@ 7,17 say " Date Scheduled Maintenance #1"
       @ 7,51 say DTOC(Mdtmx1due)
@ 8,17 say " Date Scheduled Maintenance #2"
       @ 8,51 say DTOC(Mdtmx2due)
```

```
@ 9,17 say " Date Scheduled Maintenance #3"
        @ 9,51 say DTOC(Mdtmx3due)
@ 10,14 say " Date Annual Inspection Scheduled"
        @ 10,51 say DTOC(Mdtasidue)
@ 11,19 say " Date Lube, Oil & Filter Due "
        @ 11,51 say DTOC(Mdtlofdue)
@ 12,20 say "Date Vehicle Accepted by AF "
        @ 12,51 say DTOC(Mdateacpt)
@ 13,17 say " Date Vehicle Warranty Expires"
        @ 13,51 say DTOC(Mdatewarx)
@ 14,20 say " Mileage Maintenance #1 due "
        @ 15,47 say Mmile1due
@ 15,20 say " Mileage Maintenance #2 due "
        @ 15,47 say Mmile2due
@ 16,20 say " Mileage Maintenance #3 due "
        @ 16,47 say Mmile3due
@ 17,21 say " Mileage Annual Safety Due "
        @ 17,47 say Mmileasi
@ 18,22 say " Mileage Warranty Expires"
        @ 18,47 say Mmilewarx
@ 19,16 say " Mileage Lube, Oil & Filter due"
        @ 19,47 say Mmilelof
@ 22,23 SAY "****PRESS ANY KEY TO CONTINUE****"
  WAIT ""
RETURN
******PROC #34******
PROCEDURE STOREVIMS
STORE REGNUM     TO MREGNUM
STORE TOTMILEAGE TO MTOTMILAGE
STORE VEHTYPE    TO MVEHTYPE
STORE DTMX1DUE   TO MMDTMX1DUE
STORE DTMX2DUE   TO MMDTMX2DUE
STORE DTMX3DUE   TO MMDTMX3DUE
STORE DTASIDUE   TO MDTASIDUE
STORE DATELOFDUE TO MDTLOFDUE
STORE DATEACPT   TO MDATEACPT
STORE DATEWARX   TO MDATEWARX
STORE MILE1DUE   TO MMILE1DUE
STORE MILE2DUE   TO MMILE2DUE
STORE MILE3DUE   TO MMILE3DUE
STORE MILEASI    TO MMILEASI
STORE MILEWARX   TO MMILEWARX
STORE MILELOF    TO MMILELOF
RETURN
******PROC #35******
PROCEDURE STOREVAL
STORE REGNUM   TO MREGNUM
STORE VISNUM   TO MVISNUM
STORE NOUN     TO MNOUN
STORE ASC      TO MASC
STORE VORGCODE TO MVORGCODE
STORE NUMAUTH  TO MNUMAUTH
STORE NUMASGN  TO MNUMASGN
STORE MISESSN  TO MMISESSN
```

```
STORE VALMGT   TO MVALMGT
STORE MGTISORG TO MMGTISORG
STORE MGTISASC TO MMGTISASC
STORE PRIREC   TO MPRIREC
RETURN
******PROC #36*******
PROCEDURE GETVAL
@ 0,0 CLEAR
@ 0,1 TO 24,79 DOUBLE
@ 4,2 TO 4,78 DOUBLE
@ 20,2 TO 20,78 DOUBLE
@ 1,25 SAY MTITLE
@ 3,20 SAY RTRIM(NOUNS) + " A VEHICLE (VAL DATABASE)"
@ 6,13 say "Vehicle Registration Number"
      @ 6,42 get Mregnum picture "@! 99A99999"
@ 7,14 say "Vehicle I & S Stock Number"
      @ 7,42 get Mvisnum Picture "@! XXXXXXXXXXXXXXXXX"
@ 8,20 say "Vehicle Nomenclature"
      @ 8,42 get Mnoun picture "@! XXXXXXXXXXXXXXXXXXXXXXXX"
@ 9,19 say "Allowance Source Code"
      @ 9,42 get Masc picture "@! XXXXXXX"
@ 10,23 say "Organization Code"
      @ 10,42 get Mvorgcode picture "@! XX"
@ 11,13 say "No. Authorized for this ASC"
      @ 11,42 get Mnumauth picture "9999"
@ 12,15 say "No. Assigned for this ASC"
      @ 12,42 get Mnumauth picture "9999"
@ 13,10 say "No. Vehicles Mission Essential"
      @ 13,42 get Mmisessn picture "9999"
@ 14,3 say "Vehicle Mgt. Code authorized this ASC"
      @ 14,42 get Mvalmgt picture "9999"
@ 15,9 say "Mgt Code + Stock No. + Org Code"
      @ 15,42 get Mmgtisorg picture "@!
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 16,13 say "Mgt Code + Stock No. + ASC "
      @ 16,42 get Mmgtisasc picture "@!
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
@ 17,16 say "Priority Recall Category"
      @ 17,42 get Mprirec picture "@! XX"
@ 22,14 say "****Enter Data in Blocked Areas and Press
Return****"
@ 23,19 say "***Cursor Up/Down or Right/Left to Edit***"
      read
         STORE 3 TO SWITCH
            DO ADDER WITH MREGNUM
   IF OWRITE = "Y"
      DO REPVAL
      CLOSE DATABASES
      RETURN
   ENDIF
      APPEND BLANK
      DO REPVAL
      CLOSE DATABASES
RETURN
```

```
******PROC #37******
PROCEDURE REPVAL
REPLACE REGNUM WITH MREGNUM
REPLACE VISNUM WITH MVISNUM
REPLACE NOUN   WITH MNOUN
REPLACE ASC    WITH MASC
REPLACE VORGCODE WITH MVORGCODE
REPLACE NUMAUTH  WITH MNUMAUTH
REPLACE NUMASGN  WITH MNUMASGN
REPLACE MISESSN  WITH MMISESSN
REPLACE VALMGT   WITH MVALMGT
REPLACE MGTISORG WITH MMGTISORG
REPLACE MGTISASC WITH MMGTISASC
REPLACE PRIREC   WITH MMPRIREC
CLEAR GETS
RETURN
******PROC #38*******
PROCEDURE SHOWVAL
@  0,0 CLEAR
@  0,1 TO 24,79 DOUBLE
@  4,2 TO 4,78 DOUBLE
@  20,2 TO 20,78 DOUBLE
@  2,25 SAY MTITLE
@  3,20 SAY RTRIM(NOUNS) + " A VEHICLE (VAL DATABASE)"
@  6,13 say "Vehicle Registration Number"
        @ 6,42 say Mregnum
@  7,14 say "Vehicle I & S Stock Number"
        @ 7,42 say Mvisnum
@  8,20 say "Vehicle Nomenclature"
        @ 8,42 say Mnoun
@  9,19 say "Allowance Source Code"
        @ 9,42 say Masc
@ 10,23 say "Organization Code"
        @ 10,42 say Mvorgcode
@ 11,13 say "No. Authorized for this ASC"
        @ 11,42 say Mnumauth
@ 12,15 say "No. Assigned for this ASC"
        @ 12,42 say Mnumasgn
@ 13,10 say "No. Vehicles Mission Essential"
        @ 13,42 say Mmisessn
@ 14,3  say "Vehicle Mgt. Code authorized this ASC"
        @ 14,42 say Mvalmgt
@ 15,9  say "Mgt Code + Stock No. + Org Code"
        @ 15,42 say Mmgtisorg
@ 16,13 say "Mgt Code + Stock No. + ASC"
        @ 16,42 say Mmgtisasc
@ 17,16 say "Priority Recall Category"
        @ 17,42 say Mprirec
@ 22,23 SAY "****PRESS ANY KEY TO CONTINUE****"
    WAIT ""
RETURN
******PROC #39******
PROCEDURE FINDER
    @ 0,0 CLEAR
```

```
    @ 12,22 SAY "THIS VEHICLE EXISTS IN THE DATABASE"
    @ 13,27 SAY "DO YOU WANT TO OVERWRITE ? "
        STORE "Y" TO ANSWER
    @ 13,44 GET ANSWER PICTURE "@! Y"
        IF ANSWER = "N"
           DO DBRTNS
        ENDIF
           RETURN
******PROC #40******
PROCEDURE HEADER
DO SWITCHER
@ 5,12 SAY "CURRENT DATE"
@ 5,29 SAY DATE()
@ 6,2  SAY "DATE LAST RECORD ADDED"
@ 6,29 SAY LUPDATE()
@ 7,5  SAY "REMAINING DISKSPACE"
@ 7,26 SAY DISKSPACE()
@ 5,52 SAY "CURRENT TIME"
@ 5,65 SAY TIME()
@ 6,43 SAY "# RECORDS IN DATABASE"
@ 6,65 SAY LTRIM(STR(RECCOUNT()))
@ 7,49 SAY "DATABASE IN USE"
@ 7,65 SAY DBF()
CLOSE DATABASES
RETURN
******PROC #41******
PROCEDURE AGAIN
@ 0,0 CLEAR
@ 12,21 SAY "DO YOU WISH TO SEE ANOTHER LIST ? Y/N"
  STORE "Y" TO REPEAT
@ 12,60 GET REPEAT PICTURE "@! Y"
  READ
     IF REPEAT = "N"
       @ 0,0 CLEAR
       CLOSE DATABASES
       DO MAIN
     ENDIF
       @ 0,0 CLEAR
       CLOSE DATABASES
RETURN
******PROC #42******
PROCEDURE PRINTER
@ 0,0 CLEAR
@ 12,18 SAY "DO YOU WISH TO PRINT THE LIST ? Y/N"
  STORE "Y" TO PANSWER
@ 12,59 GET PANSWER PICTURE "@! Y"
  READ
IF PANSWER = "Y"
 @ 0,0 CLEAR
 @ 12,20 SAY "MAKE SURE THE PRINTER IS ON AND ALLIGNED"
 @ 23,25 SAY "***PRESS ANY KEY WHEN READY***"
   WAIT ""
 IF SWITCH = 1
   DISPLAY FOR PKLOTNO = MPKLOTNO OFF TO PRINT
```

```
ENDIF
IF SWITCH = 2
  DISPLAY FOR PKSTATUS = MPKSTATUS OFF TO PRINT
ENDIF
IF SWITCH = 3
  DISPLAY REGNUM, DISPDEST, DISPRQDT, DISPMHRS, FOR DISPRQDT
         > MBDISPDT .AND. DISPRQDT < MEDISPDT OFF TO PRINT
ENDIF
IF SWITCH = 4
  DISPLAY FOR DISPDEST = MDISPDEST OFF TO PRINT
ENDIF
IF SWITCH = 9
  IF DTMX1DUE > CTOD(MBEGINDT) .AND. DTMX1DUE < CTOD(MENDDT)
     .OR. DTMX2DUE > CTOD(MBEGINDT) .AND. DTMX2DUE > CTOD(MBEGINDT)
     .OR. DTMX3DUE > CTOD(MBEGINDT) .AND. DTMX3DUE < CTOD(MENDDT)
     DISPLAY REGNUM, DTMX1DUE, DTMX2DUE, DTMX3DUE, MILE1DUE,
     MILE2DUE, MILE3DUE, TOTMILEAGE OFF TO PRINT
  ENDIF
ENDIF
IF SWITCH = 10
   DISPLAY REGNUM, TOTMILEAGE, VEHTYPE, DTASIDUE, MILEASI FOR
   DTASIDUE > CTOD(MBEGINDT) .AND. DTASIDUE < CTOD(MENDDT) OFF TO
   PRINT
ENDIF
IF SWITCH = 11
   DISPLAY REGNUM, TOTMILEAGE, VEHTYPE, DATELOFDUE, MILELOF
   FOR DATELOFDUE > CTOD(MBEGINDT) .AND. DATELOFDUE <
   CTOD(MENDDT) OFF TO PRINT
ENDIF
IF SWITCH = 12
   DISPLAY REGNUM, EXLSTDT, EXNXTDT, FOR EXNXTDT >
   CTOD(MBEGINDT) .AND. EXNXTDT < CTOD(MENDDT) OFF TO PRINT
ENDIF
IF SWITCH = 13
   DISPLAY REGNUM, REAPLVL, REORG, RENAMEPH, REESTRTN,
   RECASEN, RECATEG, FOR RESTAT = "O" OFF TO PRINT
ENDIF
IF SWITCH = 14
   DISPLAY REGNUM, RESTAT, REAPLVL, REORG, RENAMEPH,
   REESTRTN, RECASEN, FOR RECATEG = MRECATEG OFF TO PRINT
ENDIF
IF SWITCH = 15
   DISPLAY REGNUM, REAPLVL, RENAMEPH, RECASEN, RECATEG, FOR
   REORG = MREORG .AND. RESTAT = "O" OFF TO PRINT
ENDIF
  ENDIF
● 0,0 CLEAR
IF SWITCH = 1
   DISPLAY FOR PKLOTNO = MPKLOTNO OFF
ENDIF
IF SWITCH = 2
   DISPLAY FOR PKSTATUS = MPKSTATUS OFF
ENDIF
IF SWITCH = 3
```

```
   DISPLAY REGNUM, DISPDEST, DISPRQDT, DISPMHRS, FOR DISPRQDT
   > MBDISPDT .AND. DISPRQDT < MEDISPDT OFF
ENDIF
IF SWITCH = 4
   DISPLAY REGNUM, DISPDEST, DISPRQDT, DISPMHRS, FOR DISPDEST
   = MDISPDEST OFF
ENDIF
IF SWITCH = 9
  IF DTMX1DUE > CTOD(MBEGINDT) .AND. DTMX1DUE < CTOD(MENDDT)
     .AND. DTMX2DUE > CTOD(MBEGINDT) .OR. DTMX2DUE > CTOD(MBEGINDT)
     .AND. DTMX3DUE > CTOD(MBEGINDT) .AND. DTMX3DUE < CTOD(MENDDT)
     OFF
     DISPLAY REGNUM, DTMX1DUE, DTMX2DUE, DTMX3DUE, MILE1DUE,
     MILE2DUE, MILE3DUE, TOTMILEAGE
  ENDIF
ENDIF
IF SWITCH = 10
  DISPLAY REGNUM, TOTMILEAGE, VEHTYPE, DTASIDUE, MILEASI FOR
  DTASIDUE > CTOD(MBEGINDT) .AND. DTASIDUE < CTOD(MBEGINDT) OFF
ENDIF
IF SWITCH = 11
  DISPLAY REGNUM, TOTMILEAGE, VEHTYPE, DATELOFDUE, MILELOF
  FOR DATELOFDUE > CTOD(MBEGINDT) .AND. DATELOFDUE <
  CTOD(MENDDT) OFF
ENDIF
IF SWITCH = 12
  DISPLAY REGNUM, EXLSTDT, EXNXTDT FOR EXNXTDT >
  CTOD(MBEGINDT) .AND. EXNXTDT < CTOD(MENDDT) OFF
ENDIF
IF SWITCH = 13
  DISPLAY REGNUM, REAPLVL, REORG, RENAMEPH, REESTRTN, RECASEN
  FOR RESTAT = "O"  OFF
ENDIF
IF SWITCH = 14
  DISPLAY REGNUM, RESTAT, REAPLVL, REORG, RENAMEPH, RECASEN
  FOR RECATEG = MRECATEG  OFF
ENDIF
IF SWITCH = 15
  DISPLAY REGNUM, REAPLVL, RENAMEPH, REESTRTN, RECASEN,
  RECATEG FOR REORG = MREORG .AND. RECATEG = "O" OFF
ENDIF
  RETURN
******PROC #43******
PROCEDURE EOFTEST
 IF EOF()
   IF SWITCH = 1
      @ 23,23 SAY "NO MORE VEHICLES IN PARKING LOT " +
      LTRIM(STR(MPKLOTNO))
   ENDIF
   IF SWITCH = 2
      @ 23,23 SAY "NO MORE VEHICLES IN PARKING STATUS " +
      LTRIM(MPKSTATUS)
   ENDIF
   IF SWITCH = 3
```

```
      @ 23,22 SAY "NO MORE VEHICLES BETWEEN D+" +
      LTRIM(STR(MBDISPDT)) + " AND D+" + LTRIM(STR(MEDISPDT))
    ENDIF
    IF SWITCH = 4
      @ 23,22 SAY "NO MORE VEHICLES FOR DESTINATION " +
      LTRIM(MDISPDEST)
    ENDIF
    IF SWITCH = 9
      @ 23,11 SAY "NO MORE VEHICLES DUE FOR MAINTENANCE
      BETWEEN " + MBEGINDT " AND " + MENDDT
    ENDIF
    IF SWITCH = 10
      @ 23,29 SAY "NO MORE VEHICLES DUE FOR ASI BETWEEN " +
      MBEGINDT " AND " MENDDT
    ENDIF
    IF SWITCH = 11
      @ 23,29 SAY "NO MORE VEHICLES DUE FOR LOF BETWEEN " +
      MBEGINDT " AND " MENDDT
    ENDIF
    IF SWITCH = 12
      @ 23,9 SAY "NO MORE VEHICLES SCHEDUELED FOR EXERCISE
       BETWEEN " + MBEGINDT " AND " MENDDT
    ENDIF
    IF SWITCH = 13
      @ 23,18 SAY "THERE ARE NO MORE OPEN VEHICLE RELEASE
CASES"
    ENDIF
    IF SWITCH = 14
      @ 23,18 SAY "NO MORE VEHICLE RELEASE CASES FOR CATEGORY
      " + MRECATEG
    ENDIF
    IF SWITCH = 15
      @ 23,19 SAY "NO MORE VEHICLE RELEASE CASES FOR ORGCODE
      " + MORGCODE
    ENDIF
      @ 24,25 SAY "***PRESS ANY KEY TO CONTINUE***"
        WAIT ""
  ENDIF
RETURN
******PROC #44******
PROCEDURE DTGET
  @ 12,22 SAY "ENTER THE BEGINNING DATE (MM/DD/YY)"
    @ 12,59 GET MBEGINDT PICTURE "99/99/99"
  @ 13,22 SAY "   ENTER THE ENDING DATE (MM/DD/YY)"
    @ 13,59 GET MENDDT PICTURE "99/99/99"
      READ
      STORE CTOD(MBEGINDT) - CTOD(MENDDT) TO MDAYS
RETURN
```

Appendix B: Definition of Terms

algorithm	A recipe, or a set of clearly defined steps which defines how to do something (17:641).
bit	Stands for BInary digiT and represents the smallest unit of information in a digital computer (17:642).
byte	A unit of storage. One byte equals one character, or 8 bits, of storage (17:643).
code	The set of symbols that represents instructions to the computer (17:644).
compilation	The process of converting a high level language into a form that is executable by a computer (17:644)
cursor	The blinking character on the computer screen that represents the current position for entering data.
database	A group of logically related files or data-sets (19:148)
disk	A flat, circular disk of magnetic material on which data is recorded, referred to as a "floppy disk" or microdiskette (17:646).
file	A collection of related data (17:648).
field	Contains an item of data, for example a name (20:31).
hard disk	A rigid metal disk coated with magnetic material that is capable of storing data in much greater quantities than a floppy disk (17:649).
index	A file of key fields that is maintained in a specific order to speed access to specific records in a file.
key field	A specific used to categorize, select, or sort the records in a file.
language	A set of symbols that a programmer uses to encode instructions to the computer (17:651).

record	A collection of fields of information that describes attributes about an item, for example, name, color hair, age, color eyes describes an individual.
software	A set of instructions which is executed by a computer commonly referred to as a program (17:656).
testing	The process of using input to exercise program code in a controlled manner to expose errors (21:321).
top-down design	The process of building a software system beginning at the top of the software hierarchy and integrating all components before proceeding to the next lower level (21:321).
user friendly	A software program that is easy to use, tolerant of operator errors, easy to learn, and acknowledges that humans are imperfect creatures (22:2).
validation	A process that ensures that the software produced, accomplishes the objectives and performance attributes prescribed by its requirements specification (21:321).
Vehicle Authorization Listing (VAL)	A list of approved requirements for vehicles that specifies what vehicles (by type and number) are authorized at a base.

Appendix C: User Guide

Introduction

The WRM Vehicle Management System is an easy-to-use program designed for use by transportation personnel to manage the WRM vehicle fleet. The user need only have a basic working knowledge of microcomputers to use this program effectively. This program can be used independent of any other software system and will operate on IBM™ or IBM ™ compatible computers (this includes Zenith™ 100 or 248 systems). The printed output can be produced on any printer capable of emulating the Epson™ standard.

Installation

The WRM Vehicle Management System is easy to install on a hard disk or can be operated on a floppy disk system. To install on a hard disk follow these steps:

1. Insert WRM System Disk A into disk drive A. Ensure that the computer prompt says "A:", and type the following:

A:Install.bat

2. A subdirectory named WRM has been created on the hard disk and the contents of System Disk A has been copied to the WRM subdirectory.

3. Remove System Disk A and insert System Disk B and type copy *.* c:\WRM.

4. The contents of system disk B have now been copied to the WRM subdirectory.

Users intending to operate this program on a floppy disk

system should make backup copies of both system disks before running the program. The diskcopy function provided with the MS DOS system disks will work nicely. To execute this program you must place the system disk in the A disk drive, type diskcopy and press return. The program will ask you which drive is the source drive and which drive is the destination drive. You should review the diskcopy section of your MS DOS user's manual before attempting to run the program.

Program Operation

If you are operating a floppy based system, place the backup WRM System disk A in disk drive A and type WRM and press return. Once the Copyright notice has appeared on the screen, remove the backup System Disk A and insert the backup System Disk B. If you are operating a hard disk system, and have run the install program, you must do the following:

1. At the "A:" prompt, type C: and press return.
2. At the "C:" prompt, type cd \WRM and press return (Note: make sure to leave the space between the cd and \WRM).
3. Type WRM and press return.
4. A notice will appear on the screen, advising you that the program is being loaded.

Once the program has been loaded, a copyright notice will appear, please read the notice and comply. You may freely distribute this program within military channels for military purposes. Please do not distribute the program outside of military channels. The copyright notice is depicted in figure 1 on the following page.

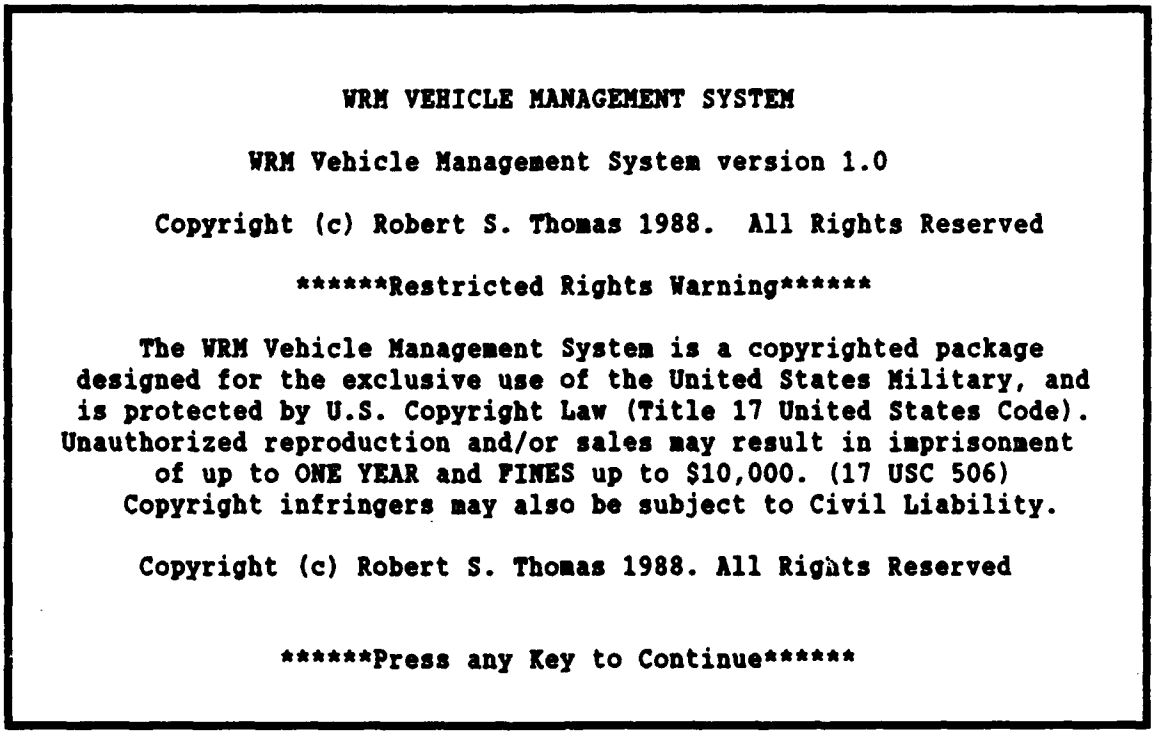


Fig. 1. Restricted Rights Warning Screen

Once you press any key to continue, the startup menu will appear as depicted in figure 2.

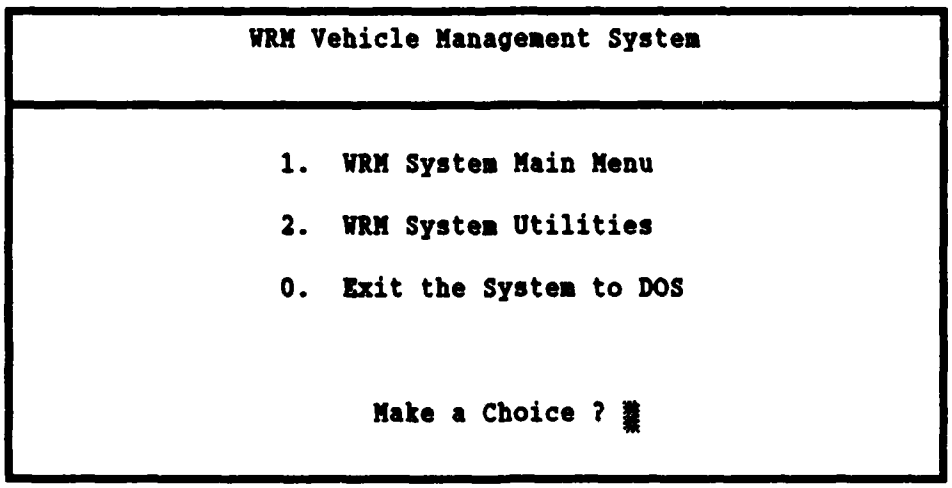


Fig. 2. Startup Menu

The majority of the activities you will perform, can be accessed by entering choice "1" and pressing return. After you have entered choice "1" the System Main Menu will appear as depicted in figure 3.

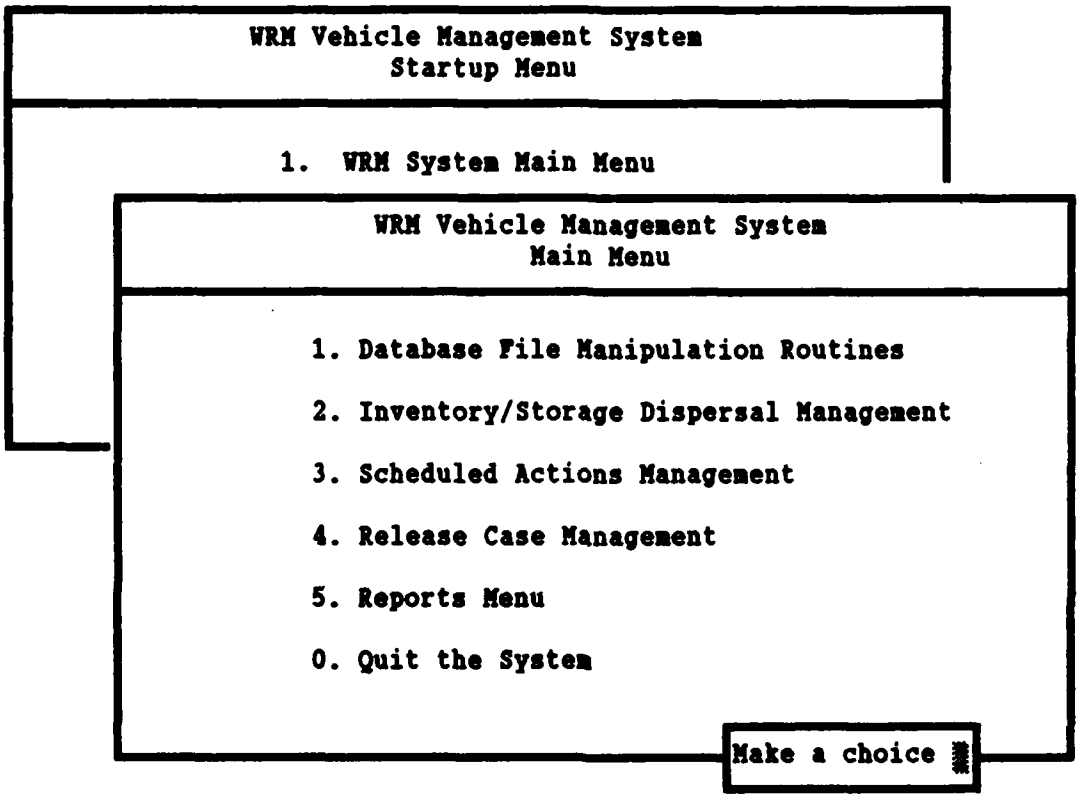


Fig. 3. WRM Main Menu Screen

Option "1" on the System Main Menu will permit you to view, add, edit, delete, or undelete data in any of the databases. Your first opportunity to use this module will begin with adding vehicles to the databases. There are eight databases that are grouped by their function. The vehicle registration number will be the first piece of vehicle data you will enter in any of the databases. The vehicle registration number will be the main way you will locate

vehicle information. Once you enter choice "1" you will see the screen depicted in figure 4.

```
                  WRM Vehicle Management System
                            Main Menu

             1. Database File Manipulation Routines

  ****THE DATABASES CONTAIN THE FOLLOWING FIELDS OF INFORMATION****

  MASTER      OLVIMS       VAL        RELEASE     MANMAINT    DISPERSAL

   REGNUM       REGNUM     REGNUM      REGNUM       REGNUM       REGNUM
  MGTCODE   TOTMILEAGE     VISNUM     REAPLVL     MXSTATUS     DISPDEST
 ASGN:MGT      VEHTYPE       NOUN      RESTAT    MXRTNDATE     DISPRQDT
   IS:NUM     DTMX1DUE        ASC       REORG       MXRMKS     DISPMHRS
      NSN     DTMX2DUE   VORGCODE    RENAMEPH   SIINTERVAL     DISPINST
  ASGN:IS     DTMX3DUE   NUM:AUTH    RERQDATE     SISWITCH
    NOMEN     DTASIDUE   NUM:ASGN     REACTDT     LOFINTER
  ORGCODE     DTLOFDUE   MIS:ESSN    REESTRTN    LOFSWITCH
   OWNCMD     DATEACPT     VALMGT     RECASEN
   USECMD     DATEWARX   MGTISORG     RECATEG
 ITEMCODE     MILE1DUE   MGTISASC      RERMKS       PKPLAN     EXERCISE
  REPCODE     MILE2DUE    PRI:REC
  USECODE     MILE3DUE                              REGNUM       REGNUM
 SPEQCODE      MILEASI                             PKLOTNO      EXLSTDT
DATE:ASGN     MILEWARX      CHOICE ?               PKROWNO      EXNXTDT
 MOBLCODE      MILELOF                             PKCOLNO       EXRMKS
      ASC                                         PKSTATUS

     **SELECT THE CAPITALIZED LETTER OF YOUR CHOICE AND PRESS RETURN**
[M]aster [O]lvims [V]al [R]elease m[A]nmaint [D]ispersal [P]kplan [E]xer
```

Fig. 4. Database Routines Menu

You will recognize the database names and perhaps the fields of information contained within them. If you have any question about a field of information in one of the databases, you need only select the database by typing the bracketed letter on the bottom line and press return. Once you press return the screen will clear and a database summary screen will appear. This screen will show that you do not have any

vehicles currently in the database. When you enter the database of your choice the appropriate summary screen will appear. Figure 5 is the summary screen for the Master database.

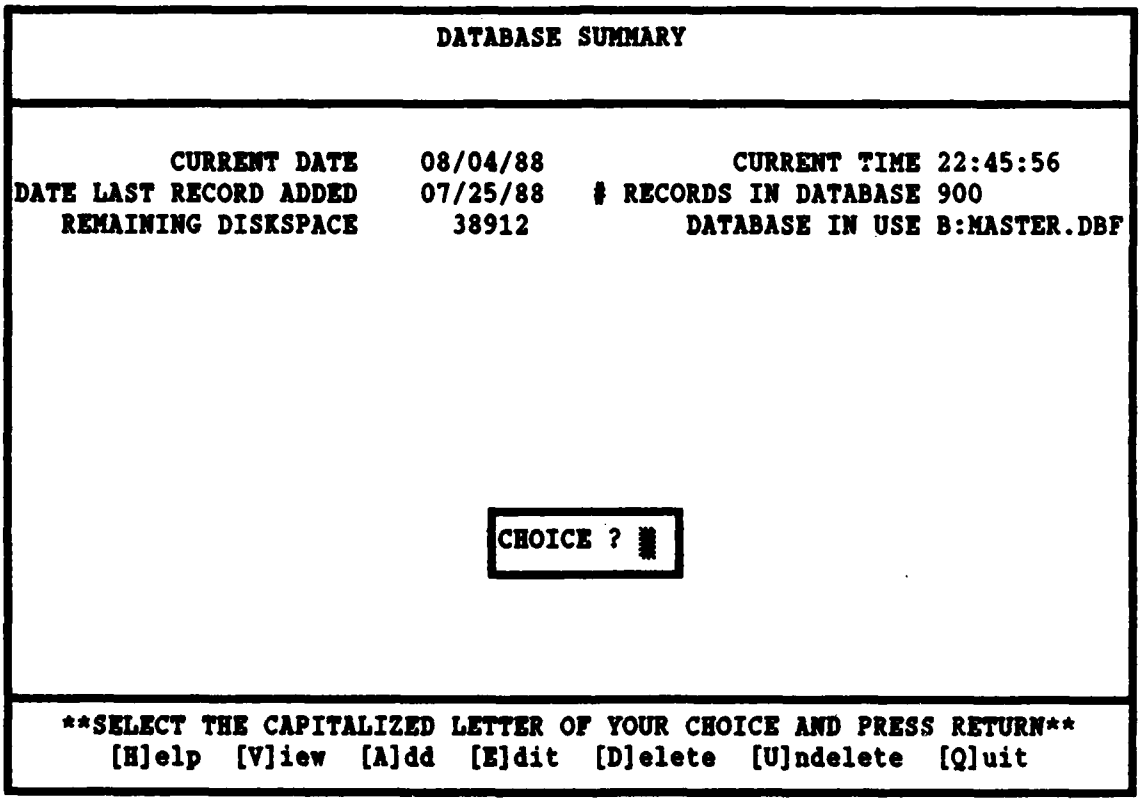


Fig. 5. Database Summary/Actions Menu

At this point you can recognize the operations that allow you to control the information contained within the databases. The options listed at the bottom are now specifically for the Master database. If you had chosen one of the other databases at figure 4, these options would now apply to that database. If you are unsure what each of the fields of information represent, you need just enter "H" for help and an explanation of all the fields will be displayed as depicted in figure 6.

DATABASE HELP SCREEN
FOR MASTER.DBF

The Master database contains the following fields information:

	Field name	Data type	width	decimal
[1]	REGNUM	CHARACTER	8	0
[2]	MGTCODE	CHARACTER	4	0
[3]	ASGNMGT	CHARACTER	4	0
[4]	ISNUM	CHARACTER	18	0
[5]	NSN	CHARACTER	18	0
[6]	ASGNIS	CHARACTER	18	0
[7]	NOMEN	CHARACTER	25	0
[8]	ORGCODE	CHARACTER	2	0
[9]	OWNCMD	CHARACTER	2	0
[10]	USECMD	CHARACTER	2	0
[11]	ITEMCODE	CHARACTER	1	0
[12]	REPCODE	CHARACTER	1	0
[13]	USECODE	CHARACTER	1	0
[14]	SPEQCODE	CHARACTER	10	0
[15]	DATEASGN	DATE	8	MM/DD/YY

****PRESS ANY KEY TO CONTINUE*****
****THE HELP FILE CONTAINS MORE INFORMATION****

Fig. 6. MASTER.DBF Help Screen

The screen begins by naming each of the fields by name, type, and length. Once you press return the abbreviated field name will be identified by its regular name. For example REGNUM will be identified as registration number. The help screens will continue as long as you press any key or until they have all been displayed for that database.

The add options will allow you to add new vehicles into the database. You must remember that each database contains different information, therefore you will need to enter the information appropriate for that vehicle in each of the databases. You may elect to not use certain options offered in the WRM program. If you select the add option the screens

appropriate database add screens will appear as in figures 7-14.

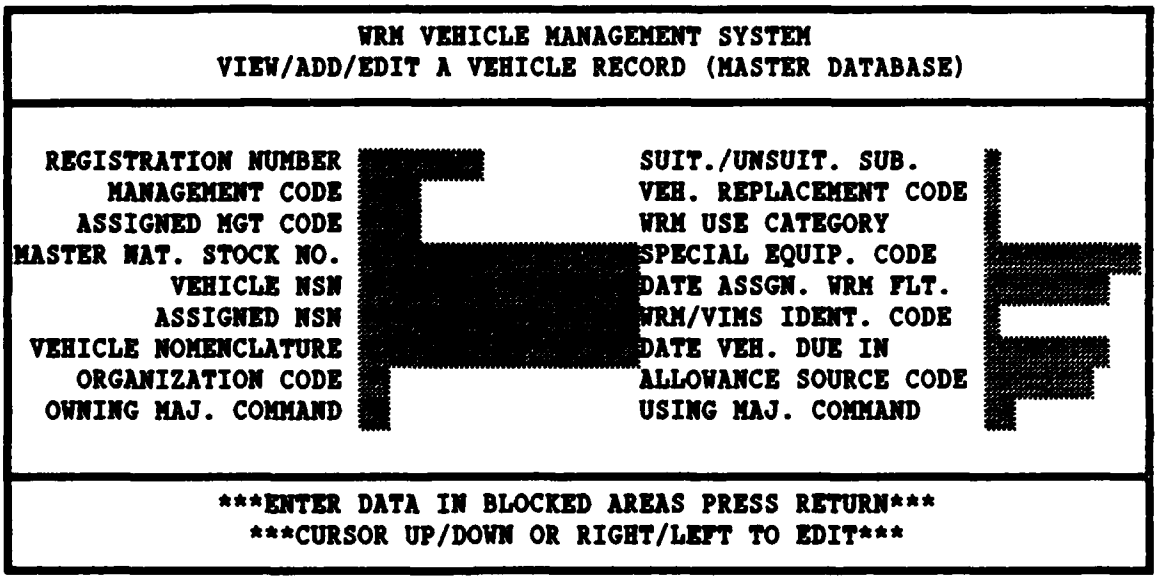


Fig. 7. Master.dbf View/Add/Edit Screen

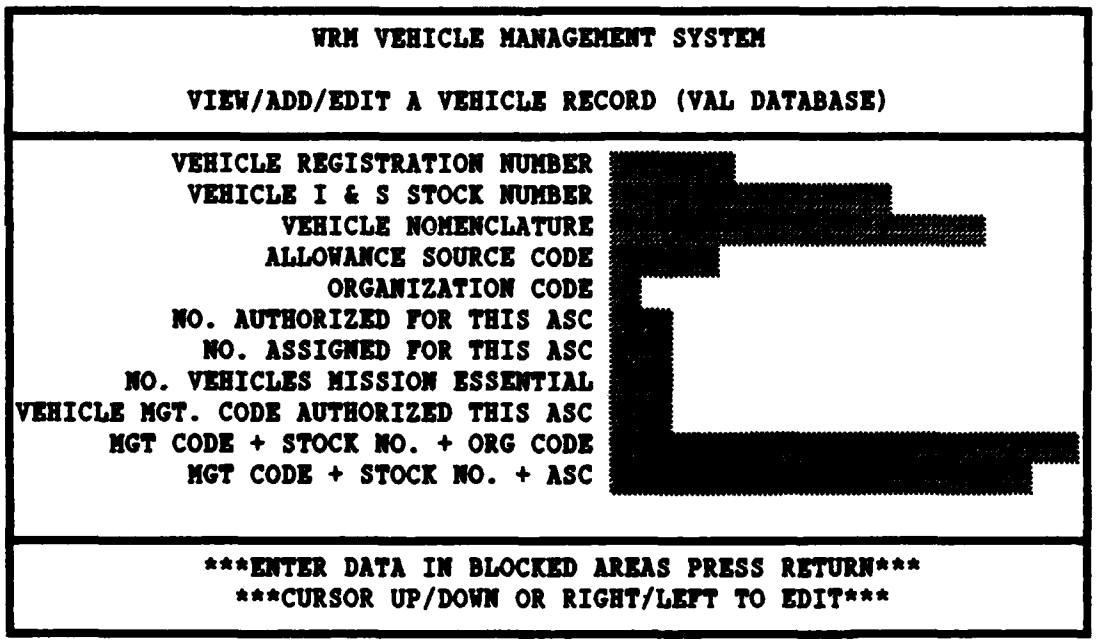


Fig. 8. VAL.DBF View/Add/Edit Screen

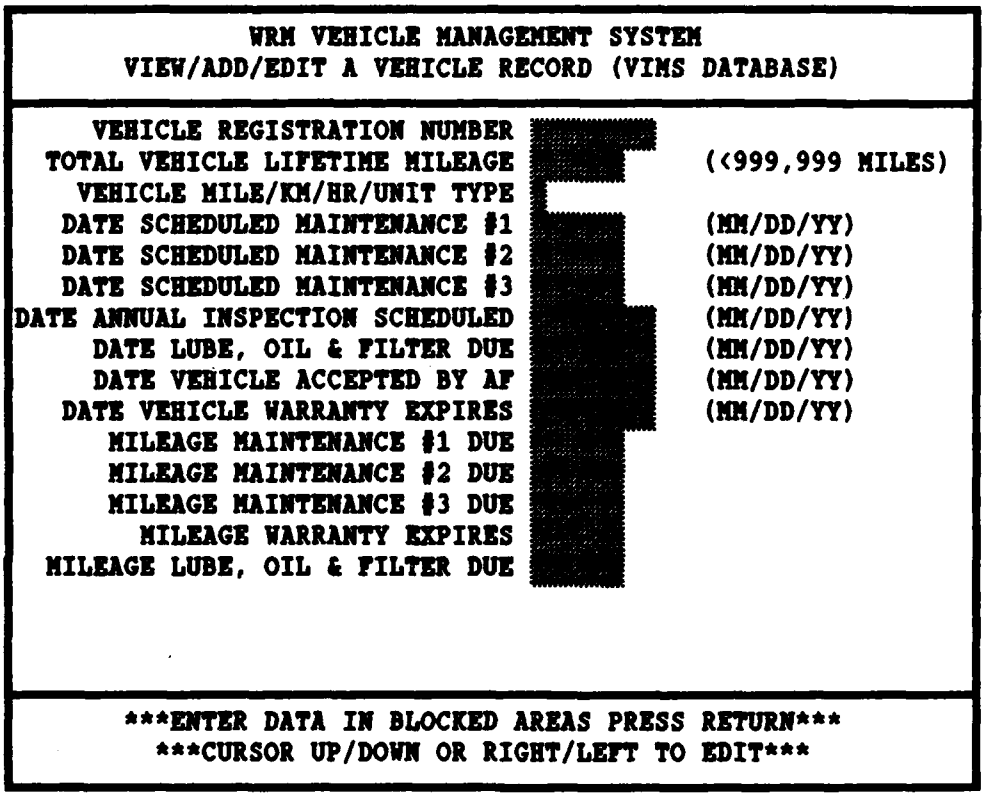


Fig. 9. VIMS.DBF View/Add/Edit Screen

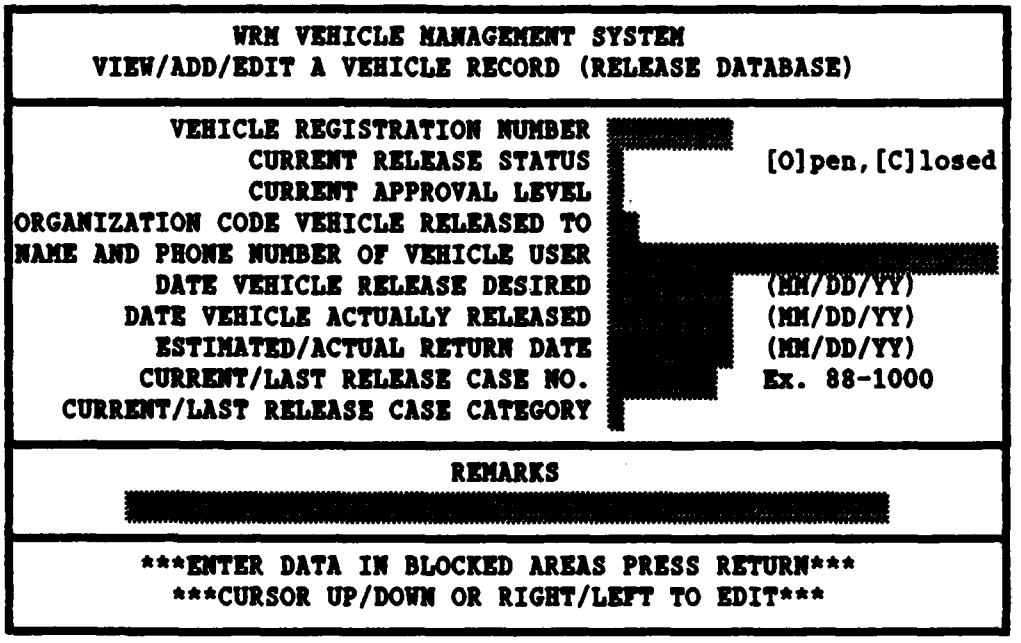


Fig. 10. RELEASE.DBF View/Add/Edit Screen

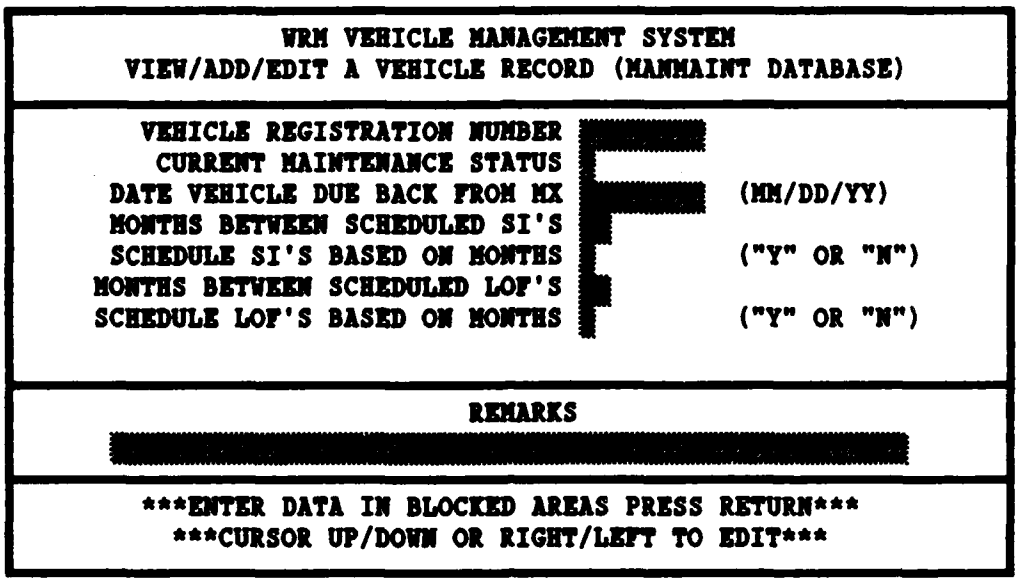


Fig. 11. MANMAINT.DBF View/Add/Edit Screen

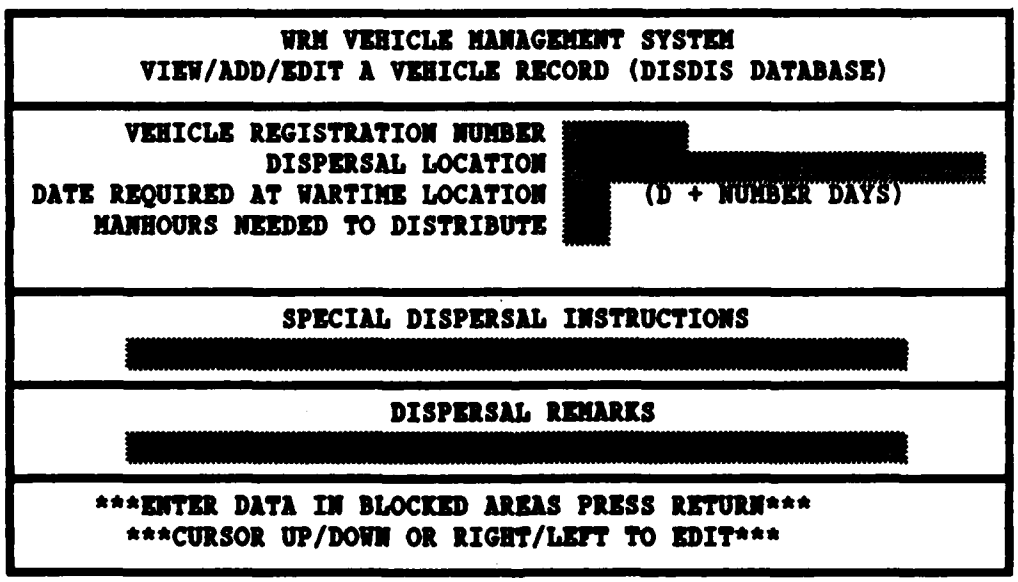


Fig. 12. DISDIS.DBF View/Add/Edit Screen

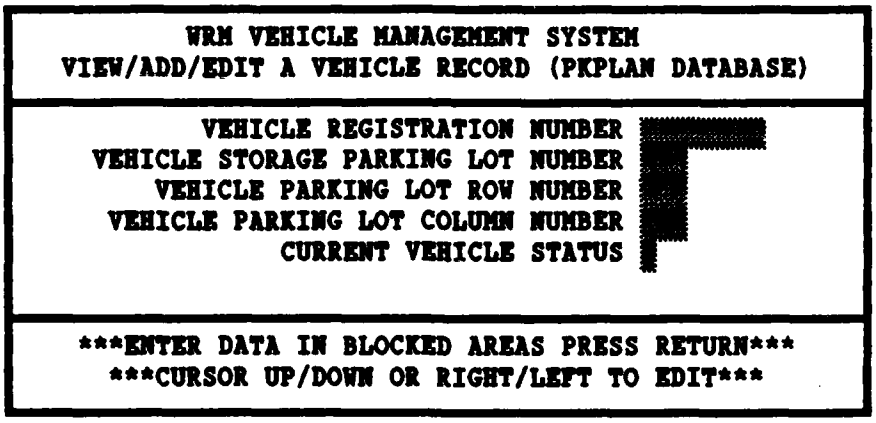


Fig. 13. PKPLAN.DBF View/Add/Edit Screen

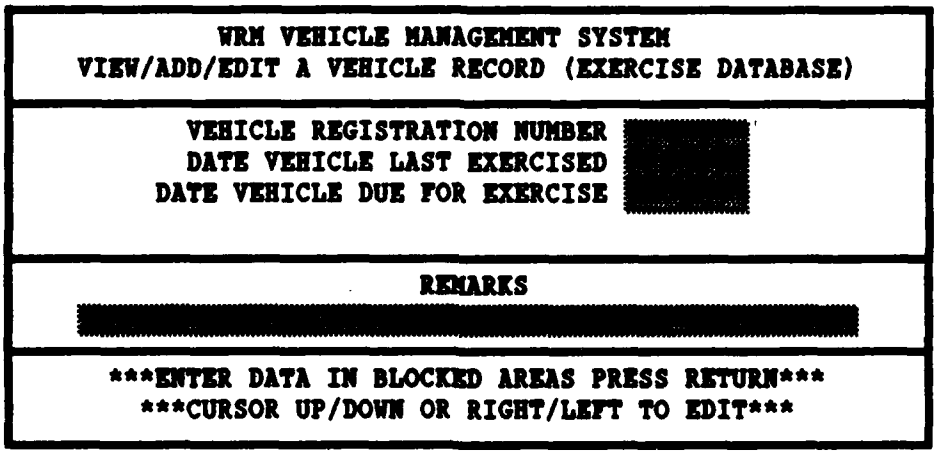


Fig. 14. EXERCISE.DBF View/Add/Edit Screen

You will notice that the view, add, and edit screens will appear the same. The only difference will be that the view and edit screens will ask you to enter a registration number, and then the database is searched for that vehicle. If it is not found a message will appear advising you that this registration number is not in the database. If the registration number is found, the vehicle data will appear in the information blocks. The view screens will not let you change data, only view it. The edit screens will allow you to

view the data and change it.

The delete and undelete functions will ask you for a vehicle registration number. Once you enter the number, if it is found a view screen will appear with the data, and a message will appear at the bottom asking you if this is the correct record. If you've made a mistake, don't panic, just answer no and the action will be aborted. If you accidentally delete a vehicle record, you can recover by choosing the undelete function and entering that vehicle registration number. When you delete a vehicle, it is copied to a temporary database identical to the one it came from. This prevents you from loosing data you didn't intend to delete. The pack module on the systems utilities menu will permanently erase vehicles in the temporary database. The database routines modules allow you to maintain control over what data enters the WRM Vehicle Management System. If you are using a hard disk system you will not have to worry about disk space. The floppy disk based system user will need to pay attention to the amount of diskspace left as presented on the database summary screen. Remember System Disk A contains the programs that run the system, not the databases. Once you have loaded the system, you can remove the A disk and keep inserting as many database disks as you need. You will need to prepare several copies of System Disk B prior to adding, editing or deleting. You can keep these organized by a range of vehicle numbers, or parking lot numbers, or by any method you choose.

Dispersal Management

The dispersal management module is entered by selecting option "2" on the System Main Menu as depicted in figure 15.

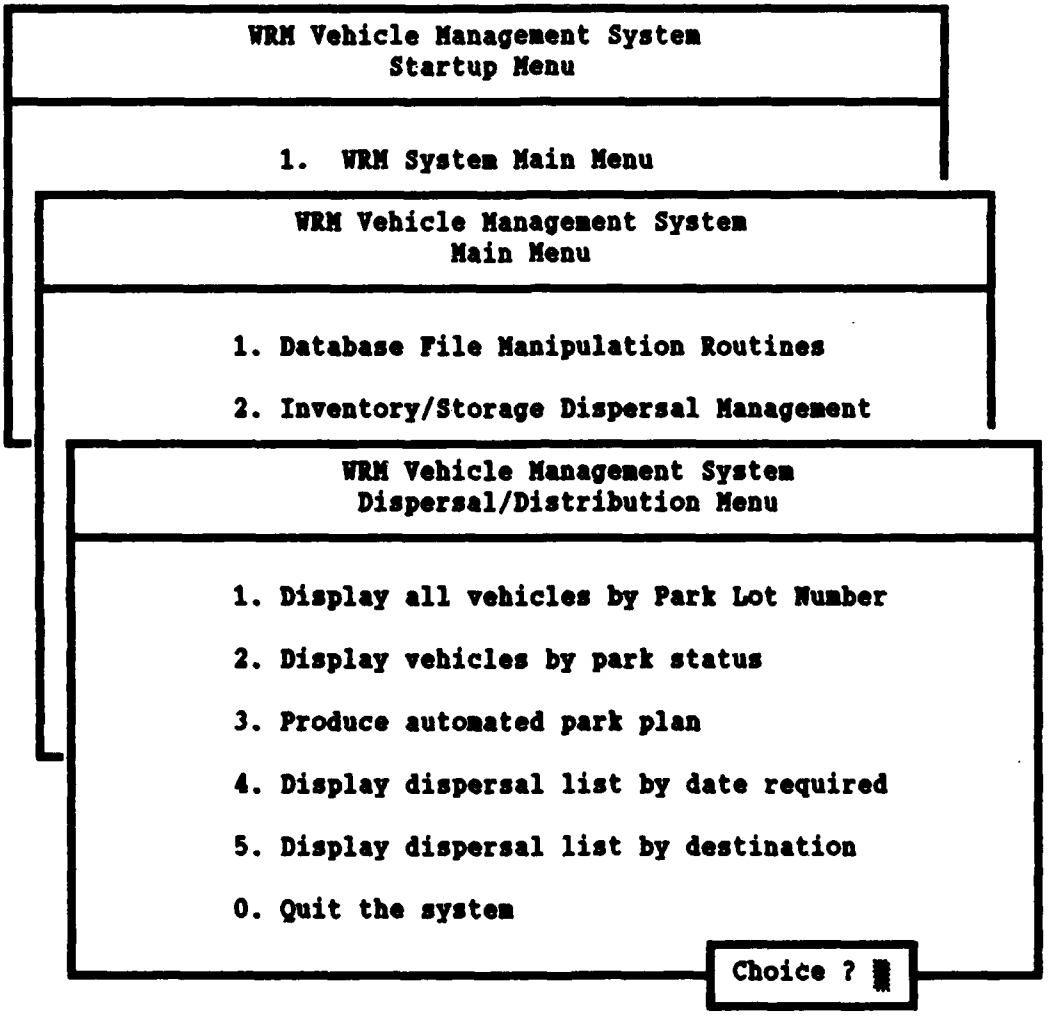


Fig. 15. Dispersal/Distribution Menu

A variety of useful options are presented in this module. This module uses the disdis and pkplan databases to aid you in maintaining inventory control during routine day to day operations or during dispersal operations. If you select option "1" or "2" on the Dispersal/Distribution Menu, you can

display all vehicles you have previously entered, by parking lot or parking status. Figure 16 depicts the parking lot display screen.

Record #	Regnum	Pklotno	Pkrowno	Pkcolno	Pkstatus
1	88B10001	1	1	4	P
2	88B10005	1	1	6	P
3	88B10007	1	1	7	M
4	88B10009	1	1	10	P
5	88B10010	1	2	1	M
7	88B05942	1	5	3	P

There are no more vehicles in parking lot 1
Press any key to continue

Fig. 16. Parking lot display screen.

Option "3" of the Dispersal /Distribution Menu will permit you to create an automated parking plan. The WRM system only produces a standardized 10 column by 20 row (200 vehicles) parking lot. At first glance this might appear limiting. If your parking lots are smaller than this size, this should not present a problem. If they are larger than the standard size, you will need to sub-divide them into several 200 vehicle lots to make effective use of this option. The advantage of being able to produce automated plans as quickly as vehicles moved far outweighs the initial discomfort associated with renumbering your existing lots. Figure 17 displays the automated parking plan that is printed.

PARKING LOT 1

ROW	COLUMN 1	2	3	4	5	6	7	8	9	10
1	88B10000 P	88B22000 P	78B32966 H	79B32967 S	86B41234 S	85B14100 P	81B38953 O	82B84213 H	75B12453 S	88B29887 P
2	82B22300 S	80B88310 P	85B41986 S	79B32968 H	81B01914 P	88B89312 P	76B81664 H	87B21459 H	83B00014 P	86B09701 H
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										

Fig. 17. Automated Parking Plan

Option "4" will allow you to view or print a dispersal list by any d+ (number of days) you enter. Figures 18 and 19 portray the display screens for this option.

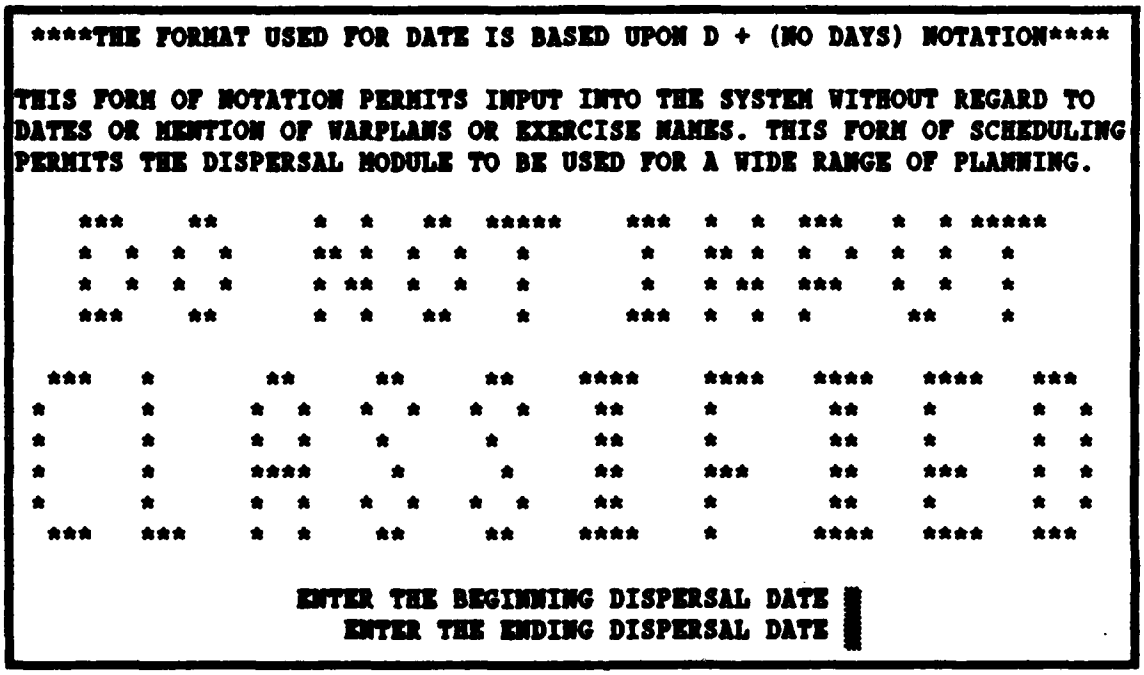


Fig. 18. Dispersal Classified Warning Screen

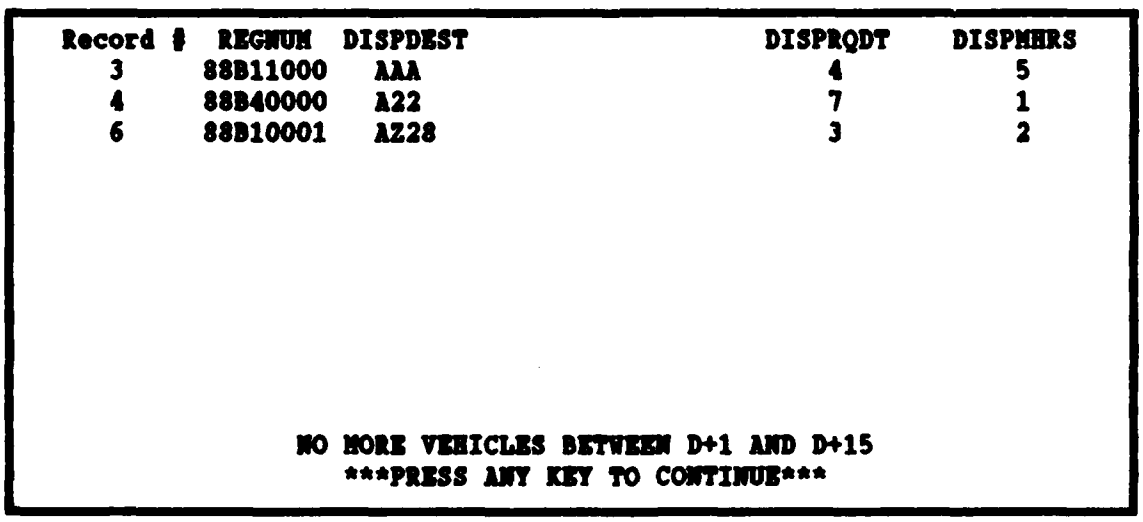


Fig. 19. Dispersal List by Date Display Screen

Option "5" will display and print the data in the same format as in figure 19 with the exception that you can select what is displayed by entering the dispersal destination code.

Scheduled Actions

The scheduled actions module is entered by selecting option "3" on the System Main Menu as is depicted in figure 20.

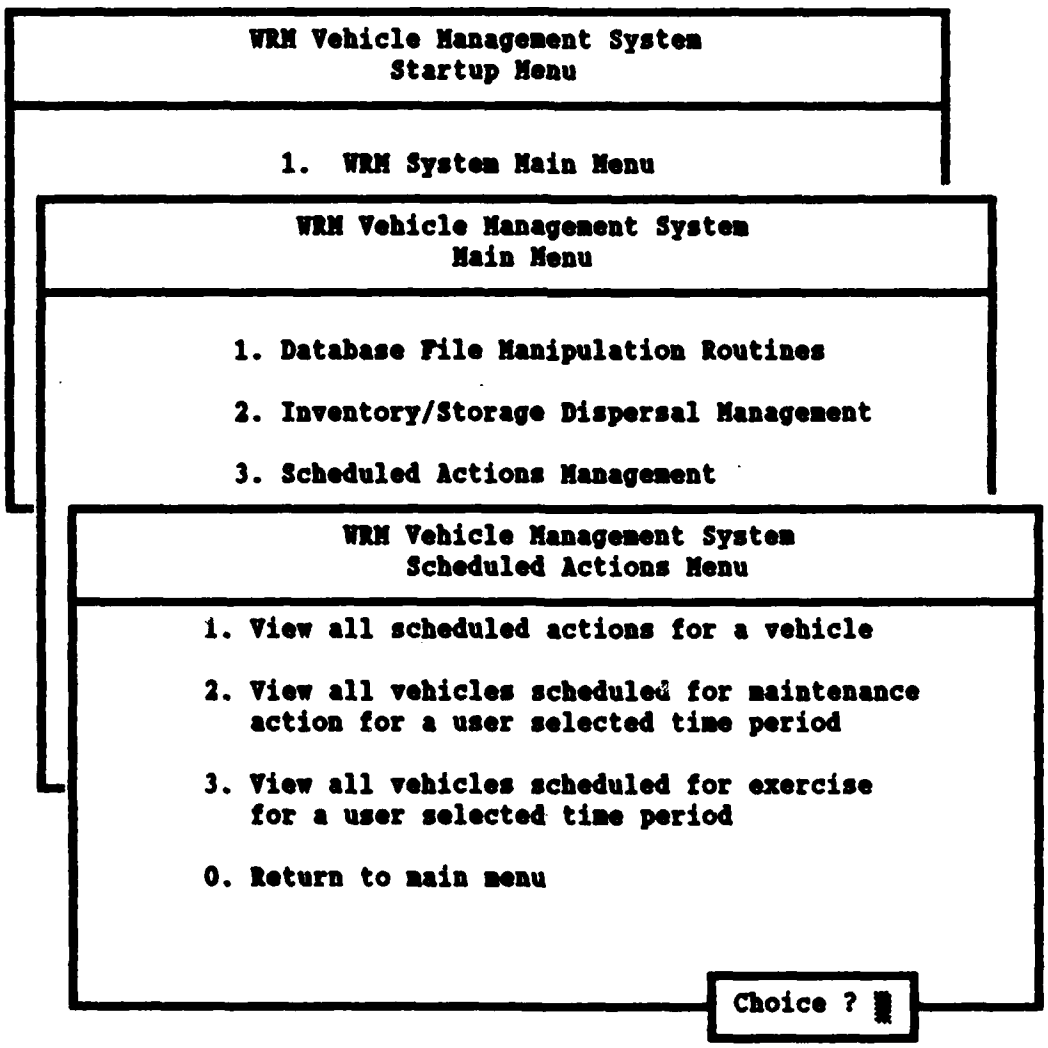


Fig. 20. Scheduled Actions Menu

Option "1" of the scheduled actions module will allow you to display all date dependent activities for a vehicle registration number you enter. Figure 21 depicts the scheduled actions view screen.

```
SCHEDULED ACTIONS REPORT FOR VEHICLE 88B10000   PK LOT# 1 ROW 2  COL 1

TOTAL VEHICLE LIFETIME MILEAGE 8700      VEHICLE MANAGEMENT CODE B101
    DATE ACCEPTED BY AIR FORCE 01/03/88  SPECIAL EQUIPMENT CODE
    DATE ASSIGNED TO WRM FLEET 03/12/88   ALLOWANCE SOURCE CODE
 WRM IDENTIFIER CODE FROM VIMS D                   VEHICLE TYPE

      RELEASE CASE INFORMATION                MAINTENANCE INFORMATION

      CURRENT RELEASE STATUS O                              DATE DUE  MILEAGE
CURRENT/LAST RELEASE CASE NO. 88-0009  SCHEDULED MAINT #1 06/15/88 8000
 DATE VEHICLE RELEASE DESIRED 04/12/88 SCHEDULED MAINT #2 12/15/88 16000
  DATE VEH. ACTUALLY RELEASED 04/14/88 SCHEDULED MAINT #3 06/15/89 22000
 ESTIMATED/ACTUAL RETURN DATE 04/22/88 ANNUAL SAFETY INSP 10/15/88 10000
                                       SCHEDULED ON NON.  IF YES Y #NON 9
       EXERCISE INFORMATION            LUBE, OIL & FILTER 06/15/88
                                       SCHEDULED ON NON.  IF YES Y #NON 0
  DATE VEHICLE LAST EXERCISED
DATE VEHICLE DUE FOR EXERCISE 09/15/88 CURRENT MAINT. STATUS
                                              DATE DUE BACK
                    REMARKS

CURRENT MAINTENANCE STATUS
   CURRENT EXERCISE STATUS
                  *****TO CONTINUE PRESS ANY KEY*****
```

Fig. 21. Scheduled Actions View Screen

If you select option "2" on the scheduled actions menu, all maintenance actions can be displayed for a time period you enter. This option can help you to display and print maintenance actions due for the WRM fleet for any time range you choose. If you choose a very narrow range, like 02/01/88 to 02/02/88, only vehicles due within that range will be displayed or printed. This option can be a very effective tool to use when you are planning to move vehicles in and out

of Vehicle Maintenance. Figure 22 depicts the display screen for this option.

REGNUM	TOTMILEAGE	DTMX1DUE	DTMX2DUE	DTMX3DUE	MILE1DUE	MILE2DUE	MILE3DUE
88B10000	7200	03/10/88	09/10/88	03/10/89	10000	15000	20000
82B32950	44000	03/17/88	09/17/88	03/17/89	47000	52000	57000
80B41200	64270	04/15/88	10/15/88	04/15/89	70000	75000	80000

NO MORE MAINTENANCE SCHEDULED BETWEEN 03/01/88 AND 05/15/88
PRESS ANY KEY TO CONTINUE

Fig. 22. Scheduled Maintenance View Screen

If you select option "3" of this module you can view all vehicles scheduled for exercise during a time period you enter. This option will allow you to schedule vehicles that need exercising to coincide with exercises. By combining the scheduled actions options with the scheduled for exercise option, you can identify scheduling conflicts that might occur when vehicles are due scheduled maintenance during the same period of time they are due for exercise. Additionally you can use the scheduled actions screen to ensure that any overlapping scheduled maintenance is performed while the vehicle is in the shop. This should reduce the overall number of hours vehicles spend needlessly in the maintenance shop.

Figure 23 portrays the scheduled for exercise screen.

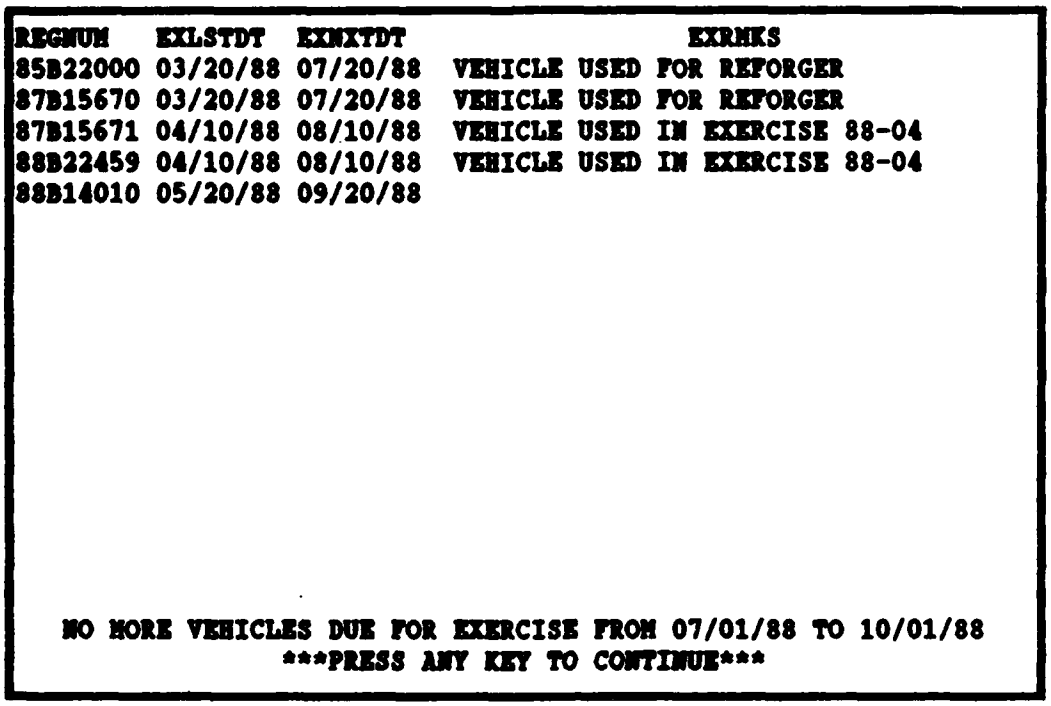


Fig. 23. Scheduled For Exercise View Screen

Release Case Management

The release case management module is designed to provide up-to-date information on open release cases, by release category, organization, or for an individual vehicle. Vehicle cases are coded "O" for open and "C" for closed. You must enter this code in the release case status block when adding or editing vehicle release case information. Select option "4" on the System Main Menu to enter the release case module as is depicted on the following page in figure 24.

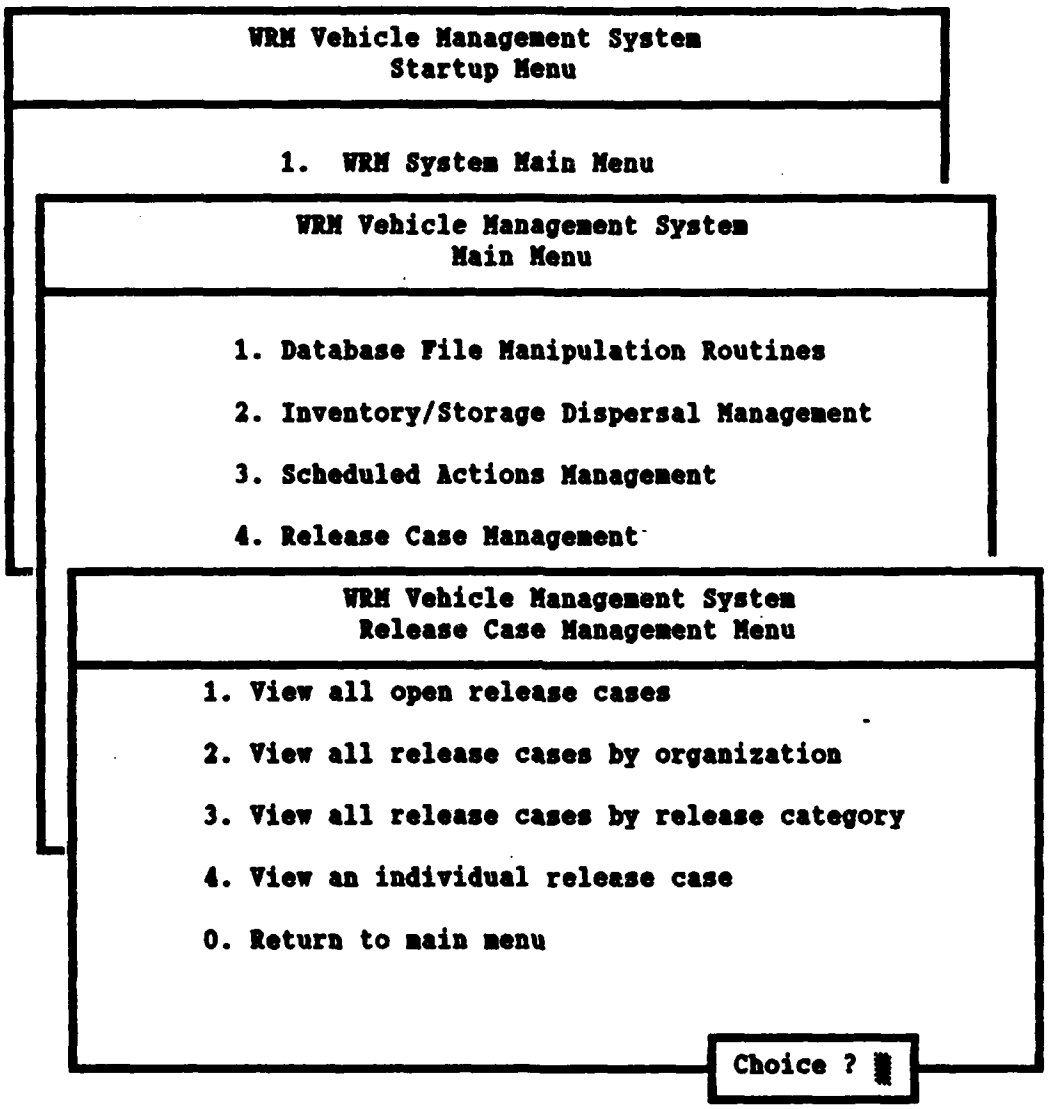
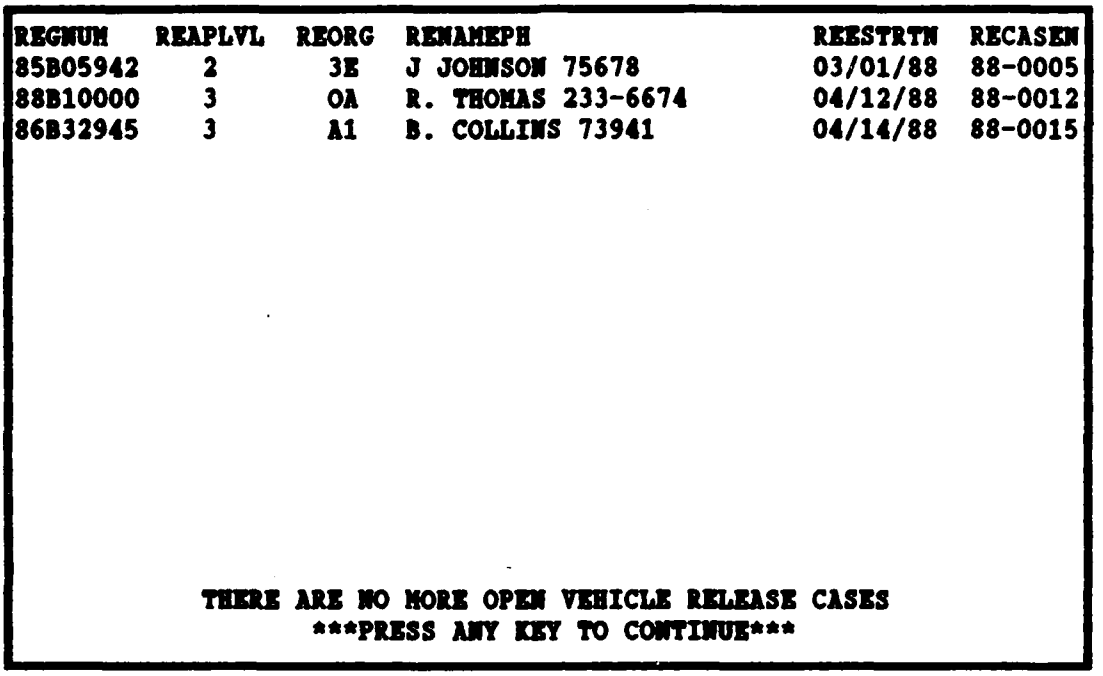


Fig. 24. Release Case Management Menu

Each of the options presents the release case information arranged in a somewhat different fashion. Each option will permit you to print or view the data. Option "4" will permit you to view all the release case data on any vehicle in the database. Figures 25 - 28 portray the view screens associated

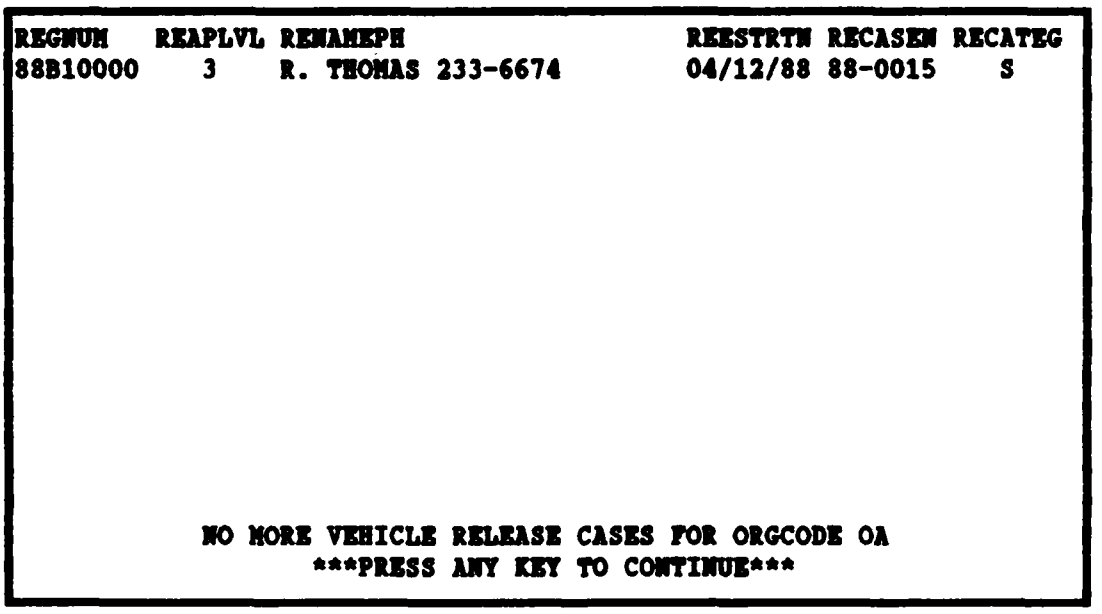
with options 1 - 4 respectively.



REGNUM	REAPLVL	REORG	RENAMEPH	REESTRTN	RECASEN
85B05942	2	3E	J JOHNSON 75678	03/01/88	88-0005
88B10000	3	OA	R. THOMAS 233-6674	04/12/88	88-0012
86B32945	3	A1	B. COLLINS 73941	04/14/88	88-0015

THERE ARE NO MORE OPEN VEHICLE RELEASE CASES
PRESS ANY KEY TO CONTINUE

Fig. 25. Open Release Case View Screen



REGNUM	REAPLVL	RENAMEPH	REESTRTN	RECASEN	RECATEG
88B10000	3	R. THOMAS 233-6674	04/12/88	88-0015	S

NO MORE VEHICLE RELEASE CASES FOR ORGCODE OA
PRESS ANY KEY TO CONTINUE

Fig. 26. Release Case by Organization View Screen

```
REGNUM    RESTAT REAPLVL RENAMEPH                    REESTRTN RECASEN
88B10000    O       3    R. THOMAS 233-6674          04/12/88 88-0015









            NO MORE VEHICLE RELEASE CASES FOR CATEGORY S
                  ***PRESS ANY KEY TO CONTINUE***
```

Fig. 27. Release Case by Release Category View Screen

```
                 WRM VEHICLE MANAGEMENT SYSTEM
              VIEW A VEHICLE RECORD (RELEASE DATABASE)

          VEHICLE REGISTRATION NUMBER 88B10000
               CURRENT RELEASE STATUS O          [O]pen,[C]losed
                CURRENT APPROVAL LEVEL 3
ORGANIZATION CODE VEHICLE RELEASED TO OA
NAME AND PHONE NUMBER OF VEHICLE USER T. Howard 73957
         DATE VEHICLE RELEASE DESIRED 04/10/88  (MM/DD/YY)
       DATE VEHICLE ACTUALLY RELEASED 04/10/88  (MM/DD/YY)
         ESTIMATED/ACTUAL RETURN DATE 04/14/88  (MM/DD/YY)
        CURRENT/LAST RELEASE CASE NO. 88-0052   Ex. 88-1000
   CURRENT/LAST RELEASE CASE CATEGORY E

                           REMARKS
         VEHICLE RELEASED TO SATISFY EMERGENCY REQUIREMENT

                 ***PRESS ANY KEY TO CONTINUE***
```

Fig. 28. Vehicle Release Case View Screen

Reports Module

The Reports module produces a variety of printed reports. The Reports module can be selected by choosing option "5" on the System Main Menu as depicted in figure 29.

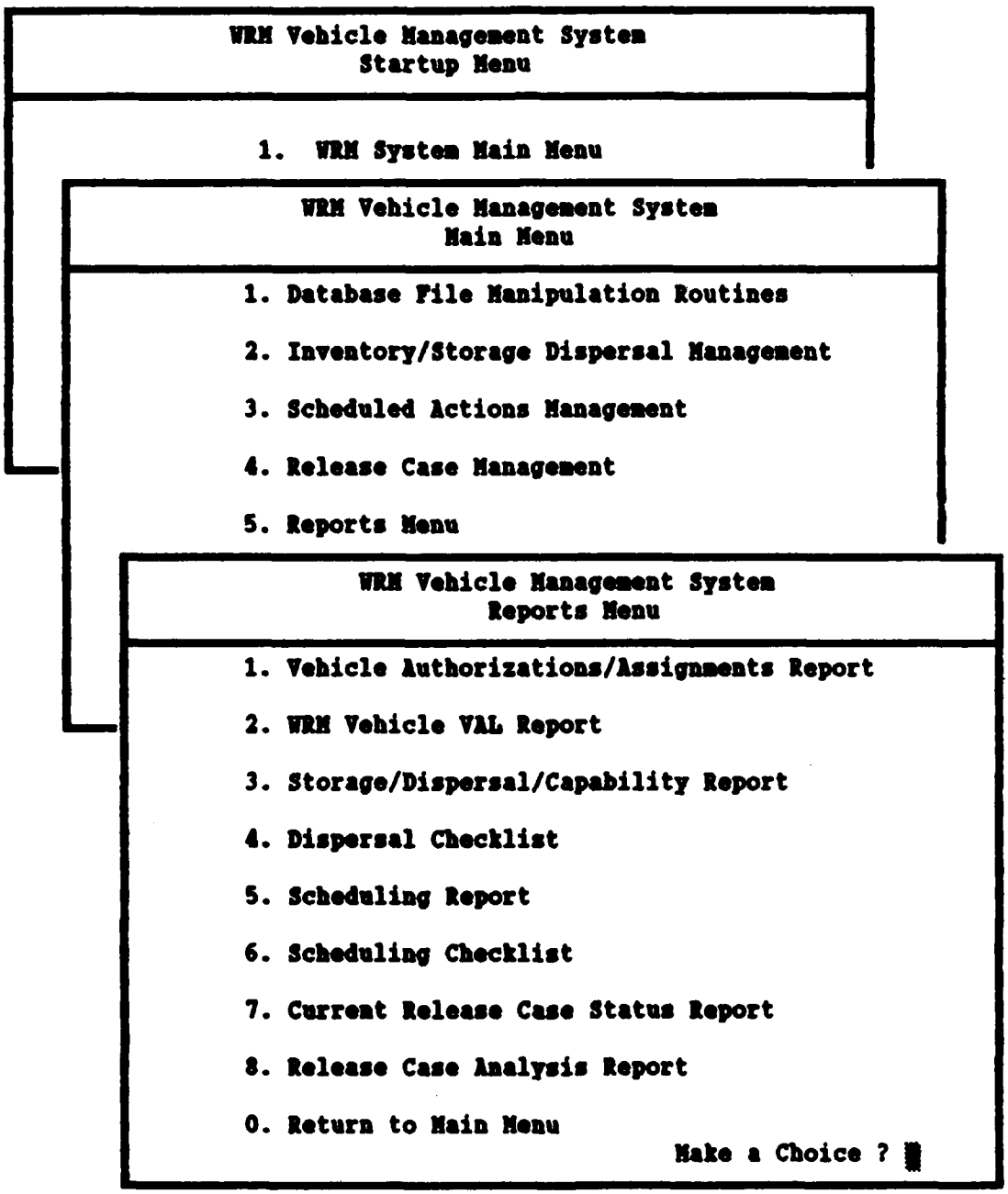


Fig. 29. Reports Menu View Screen

Figures 30 - 38 portray the various reports printed by choosing the various options in the reports module.

```
                    Vehicle Authorizations/Assignments
                          Report as of 05/10/88

                    VEHICLE:78B01645

        MGTCODE B200                          ITEM CODE S
   ASGN MGTCODE B200                           REP CODE G
     MASTER NSN 2320-01-124-7517               USE CODE M
    VEHICLE NSN 2320-01-009-6194              DATE ASGN 03/10/78
       ASGN NSN 2320-00-999-9999              MOBL CODE J
   NOMENCLATURE TRK PU CMPT 4X2                     ASC 010A00C
        ORGCODE 4N                        NO EXS TYEAR 3
        OWN CMD 0S                        NO REL TYEAR 6
        USE CMD 0S                        REL DAYS TOT 41
-------------------------------------------------------------------------
                    VEHICLE:77E00158

        MGTCODE E832                          ITEM CODE S
   ASGN MGTCODE E820                           REP CODE N
     MASTER NSN 3930-010032-3026               USE CODE M
    VEHICLE NSN 2320-01-009-6194              DATE ASGN 07/11/77
       ASGN NSN 3930-00-879-2157              MOBL CODE J
   NOMENCLATURE FKLFT ACFT LOADR                    ASC 010A00D
        ORGCODE 20                        NO EXS TYEAR 1
        OWN CMD 0H                        NO REL TYEAR 0
        USE CMD 0H                        REL DAYS TOT 0
-------------------------------------------------------------------------
                    VEHICLE:78B01647

        MGTCODE B200                          ITEM CODE S
   ASGN MGTCODE B200                           REP CODE G
     MASTER NSN 2320-01-124-7517               USE CODE M
    VEHICLE NSN 2320-01-009-6194              DATE ASGN 03/10/78
       ASGN NSN 2320-00-999-9999              MOBL CODE J
   NOMENCLATURE TRK PU CMPT 4X2 45                  ASC 010A00C
        ORGCODE 6A                        NO EXS TYEAR 2
        OWN CMD 0H                        NO REL TYEAR 3
        USE CMD 0H                        REL DAYS TOT 8
-------------------------------------------------------------------------
                              PAGE 1
```

Fig. 30. Vehicle Authorizations/Assignments Report

WRM Vehicle Authorization Listing
Report as of 05/10/88

VEHICLE:87B05550

```
VAL STOCK NO 2320-01-009-6194          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                       NUMBER AUTH 6
VAL ORG CODE 20                            NUMBER ASGN 6
  PRI RECALL 1
```

VEHICLE:78B01645

```
VAL STOCK NO 2320-01-124-7517          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                       NUMBER AUTH 6
VAL ORG CODE 20                            NUMBER ASGN 6
  PRI RECALL 2
```

VEHICLE:76B02387

```
VAL STOCK NO 2320-01-009-6194          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                       NUMBER AUTH 6
VAL ORG CODE 20                            NUMBER ASGN 6
  PRI RECALL 1
```

VEHICLE:78B01647

```
VAL STOCK NO 2320-01-124-7517          NUMBER MIS ESSN 1
NOMENCLATURE TRK PU CMPT 4X2             VAL MGT CODE B200
         ASC 010A00C                       NUMBER AUTH 6
VAL ORG CODE 20                            NUMBER ASGN 6
  PRI RECALL 1
```

PAGE 1

Fig. 31. WRM Vehicle Authorization Listing Report

Storage/Dispersal/Capability
Report as of 05/10/88

VEHICLE:78B01645

PK LOT NO: 2 PK LOT ROW: 2 PK LOT COL: 4 PARKING STATUS: S

MGTCODE	B200	REP CODE	G
VEHICLE NSN	2230-01-124-7517	USE CODE	M
NOMENCLATURE	TRK PU CMPT 4X2	DATE ASGN	03/10/78
ORG CODE	4N	MOBL CODE	J
OWN CMD	0S	ASC	010A00C
USE CMD	0S	TOT MILEAGE	52000
ITEM CODE	S	VEH TYPE	M

DISPERSAL INFORMATION

DISP DEST AZ322 DISP RQD DATE: 6
DISP MHRS 3
DISP INST:
DISP RMKS:

VEHICLE:77E00158

PK LOT NO: 1 PK LOT ROW: 4 PK LOT COL: 4 PARKING STATUS: P

MGTCODE	E832	REP CODE	S
VEHICLE NSN	3930-010032-3026	USE CODE	N
NOMENCLATURE	FKLFT ACFT LOADR	DATE ASGN	M
ORG CODE	20	MOBL CODE	J
OWN CMD	0H	ASC	010A00D
USE CMD	0H	TOT MILEAGE	
ITEM CODE	S	VEH TYPE	H

DISPERSAL INFORMATION

DISP DEST AN23 DISP RQD DATE: 1
DISP MHRS 5
DISP INST: MUST DISPERSE WITH 463L TINES
DISP RMKS:

PAGE 1

Fig. 32. Storage/Dispersal/Capability Report

DISPERSAL CHECKLIST
AS OF 05/10/88

VEHICLE REGISTRATION NUMBER	78B01645	
VEHICLE MANAGEMENT CODE	B200	
ORGANIZATION CODE	6A	
OWNING MAJOR COMMAND	0H	
USING MAJOR COMMAND	0H	
WRM VEHICLE USE CODE	M	
WRM IDENTIFIER FROM VIMS	J	
ALLOWANCE SOURCE CODE	010A00C	
VEHICLE MILE/KM/HR/UNIT TYPE	M	
PARKING LOT NUMBER	3	
PARKING ROW NUMBER	5	
PARKING COLUMN NUMBER	2	
PARKING STATUS	P	
DISPERSAL DESTINATION	AZ1522	
DATE REQUIRED AT WAR TIME LOCATION	21	(D+DAY FORMAT)
MANHOURS NEEDED TO DISTRIBUTE	5	

DISPERSAL REMARKS

SPECIAL DISPERSAL INSTRUCTIONS
VEHICLE MUST BE EQUIPPED WITH PINTLE HOOK WHEN DISPERSED

Fig. 33. Dispersal Checklist

```
SCHEDULED ACTIONS REPORT FOR VEHICLE 88B10000   PK LOT# 1 ROW 2  COL 1

TOTAL VEHICLE LIFETIME MILEAGE 8700     VEHICLE MANAGEMENT CODE B101
    DATE ACCEPTED BY AIR FORCE 01/03/88  SPECIAL EQUIPMENT CODE
    DATE ASSIGNED TO WRM FLEET 03/12/88   ALLOWANCE SOURCE CODE
 WRM IDENTIFIER CODE FROM VIMS D                  VEHICLE TYPE

      RELEASE CASE INFORMATION              MAINTENANCE INFORMATION

       CURRENT RELEASE STATUS O                              DATE DUE MILEAGE
CURRENT/LAST RELEASE CASE NO. 88-0009  SCHEDULED MAINT #1 06/15/88 8000
 DATE VEHICLE RELEASE DESIRED 04/12/88 SCHEDULED MAINT #2 12/15/88 16000
  DATE VEH. ACTUALLY RELEASED 04/14/88 SCHEDULED MAINT #3 06/15/89 22000
 ESTIMATED/ACTUAL RETURN DATE 04/22/88 ANNUAL SAFETY INSP 10/15/88 10000
                                       SCHEDULED ON MON.  IF YES Y #MON 9
       EXERCISE INFORMATION            LUBE, OIL & FILTER 06/15/88
                                       SCHEDULED ON MON.  IF YES Y #MON 0
  DATE VEHICLE LAST EXERCISED
DATE VEHICLE DUE FOR EXERCISE 09/15/88 CURRENT MAINT. STATUS
                                              DATE DUE BACK
                          REMARKS

CURRENT MAINTENANCE STATUS
   CURRENT EXERCISE STATUS
```

```
  SCHEDULED ACTIONS REPORT FOR VEHICLE 85B30569   PK LOT# 1 ROW 2  COL 2

TOTAL VEHICLE LIFETIME MILEAGE 19600    VEHICLE MANAGEMENT CODE B200
    DATE ACCEPTED BY AIR FORCE 01/03/88  SPECIAL EQUIPMENT CODE
    DATE ASSIGNED TO WRM FLEET 03/12/88   ALLOWANCE SOURCE CODE 010A00C
 WRM IDENTIFIER CODE FROM VIMS J                  VEHICLE TYPE M

      RELEASE CASE INFORMATION              MAINTENANCE INFORMATION

       CURRENT RELEASE STATUS C                              DATE DUE MILEAGE
CURRENT/LAST RELEASE CASE NO. 87-0018  SCHEDULED MAINT #1 06/15/87 18000
 DATE VEHICLE RELEASE DESIRED          SCHEDULED MAINT #2 12/15/87 24000
  DATE VEH. ACTUALLY RELEASED 03/01/87 SCHEDULED MAINT #3 06/15/88 30000
 ESTIMATED/ACTUAL RETURN DATE 04/11/87 ANNUAL SAFETY INSP 10/15/87 15000
                                       SCHEDULED ON MON.  IF YES N  #MON
       EXERCISE INFORMATION            LUBE, OIL & FILTER           18000
                                       SCHEDULED ON MON.  IF YES N #MON 0
  DATE VEHICLE LAST EXERCISED 03/10/87
DATE VEHICLE DUE FOR EXERCISE 03/10/88 CURRENT MAINT. STATUS
                                              DATE DUE BACK
                          REMARKS

CURRENT MAINTENANCE STATUS
   CURRENT EXERCISE STATUS

                          PAGE 1
```

Fig. 34. Scheduling Report

Scheduling Checklist
as of 05/10/88

VEHICLE:85B30569

VEHICLE MANAGEMENT CODE:	B200
SPECIAL EQUIPMENT CODE:	AZZ09876ZZ
VEHICLE MILE/KM/HR/UNIT TYPE:	M
DATE SCHEDULED MAINTENANCE #1:	18000
DATE SCHEDULED MAINTENANCE #2:	24000
DATE SCHEDULED MAINTENANCE #3:	30000
DATE ANNUAL INPSECTION SCHED:	08/15/88
DATE LUBE, OIL & FILTER DUE:	08/15/88
TOTAL VEHICLE LIFETIME MILEAGE:	21000
MILEAGE MAINTENANCE #1 DUE:	18000
MILEAGE MAINTENANCE #2 DUE:	24000
MILEAGE MAINTENANCE #3 DUE:	30000
CURRENT MAINTENANCE STATUS:	C
DATE VEHICLE DUE BACK FROM MAINT:	
CURRENT RELEASE CASE STATUS:	C
ESTIMATED/ACTUAL RETURN DATE:	
DATE VEHICLE LAST EXERCISED:	03/10/87
DATE VEHICLE DUE FOR EXERCISE:	03/10/88
PARKING LOT NUMBER:	1
PARKING ROW NUMBER:	2
PARKING COLUMN NUMBER:	2
PARKING STATUS:	P

Fig. 35. Scheduling Checklist

CURRENT RELEASE CASE STATUS
REPORT AS OF 05/10/88

VEHICLE IDENTIFICATION INFORMATION

REGNUM	MGT CODE	NOMENCLATURE	ORG CODE	OWN CMD	USE CMD	SPEQCODE	MOBL CODE	ASC
85B05942	B204	PASSENGER VAN 9 PAX	4M	0S	0S	AZZ09876ZZ	J	010A00C

RELEASE CASE INFORMATION

CASE NUMBER	USER NAME, PHONE NUMBER	USER ORG	RQD DATE	ACCPT DATE	EST RTN
88-0050	R JOHNSON 75678	3E	04/11/88	04/11/88	05/27/88

REMARKS:

PARKING INFORMATION

LOT NUMBER	ROW NUMBER	COLUMN NUMBER	PARK STATUS
1	5	3	P

VEHICLE IDENTIFICATION INFORMATION

REGNUM	MGT CODE	NOMENCLATURE	ORG CODE	OWN CMD	USE CMD	SPEQCODE	MOBL CODE	ASC
85b30569	B200	TRK PU CMPT 4X2	4N	0S	0S	AZZ09876ZZ	J	010A00C

RELEASE CASE INFORMATION

CASE NUMBER	USER NAME, PHONE NUMBER	USER ORG	RQD DATE	ACCPT DATE	EST RTN
88-0052	B. LANDRY 74389	3E	05/18/88	06/20/88	06/20/88

REMARKS:

PARKING INFORMATION

LOT NUMBER	ROW NUMBER	COLUMN NUMBER	PARK STATUS
2	2	4	T

Fig. 36. Current Release Case Report

Release Case Analysis Report
as of 06/18/88

ARCHIVE	MGT CODE	ORG CODE	OWN CMD	USE CMD	SPEQ CODE	ASC	REL APV LVL	REL THIS YEAR	REL TOT. DAYS	DAYS "D" CAT1	DAYS "E" CAT2	DAYS "I" CAT3	DAYS "S" CAT4	DAYS "L" CAT5	DAYS "E" CAT6	HUMD "D" CAT1	HUMD "E" CAT2	HUMD "I" CAT3	HUMD "S" CAT4	HUMD "L" CAT5	HUMD "E" CAT6
88B10000	B101	61	0E	0E		0101000	3	4	11		3	8					2	2			
78B01645	B200	4E	0S	0S		0101000	2	1	4			4						1			
77B00158	E832	20	0E	0E		0101000	3	2	5		1	4					1	1			
						TOTALS	8	7	20		4	16					3	4			

PAGE 1

Fig. 37. Release Case Analysis Report

The Reports offered in this module are generally useful for analyzing and comparing the whole fleet. The scheduling and dispersal checklists are useful during contingency exercises.

System Utilities

The system utilities module provides the options to read VIMS and VAMS data automatically into the WRM System. The VIMS data disks must be prepared exactly as described in the VAMS user manual. Normally, the fleet management section can

provide the data disks in the correct format. If it becomes necessary to download your own files than the directions for the VAMS system will produce a data disk that can be used by the WRM system.

Data can also be used from the VAMS system through the utilities module. This has the added advantage of already having been reconciled with the VIMS system.

The Pack/Index option can be used to delete vehicles from the temporary databases, and to repair damaged index files. If a database operation is inadvertently interrupted, the database index in use may be damaged. If this occurs place the disk with the damaged index in disk drive A and pack/index the disk. All database index files contained on the disk will be reindexed.

The copy option is provided for the hard disk system as it will apportion so many database records per floppy disk to provide the ability to back up databases. It is recommended that all data be backed up weekly

Bibliography

1. Department of the Air Force. Functions and Basic Doctrine of the United States Air Force. AFM 1-1. Washington: HQ USAF, 1 March 1984.

2. Department of the Air Force. Combat Support Doctrine. AFM 2-15. Washington: HQ USAF, 13 December 1985.

3. Tilford, Earl H. and William A. Buckingham. "The Limits of Superiority: Air Power in Vietnam," USAF and Force Employment-Lessons Learned from Air Power. SOS Course 22D, Area IV-Phase IV, Gunter AFS AL, 45-51 (August 1984).

4. Peppers, Jerome G. "A Brief Historical Review of U.S. Military Logistics 1939-1985," Military Logistics. Air Force Institute of Technology, 6-112 (1987).

5. Rowden, Vice Admiral William H. U.S. Navy. "Strategic Sealift and the Merchant Marine," Defense 85, 6: 15-21 (July 85).

6. Government Accounting Office. Responsiveness of the Civil Reserve Air Fleet can be Improved. Report to Congress; GAO/NSIAD-86-47. Washington: Government Printing Office, March 1986.

7. Air Force Logistic Command. World Wide Vehicle Requirements Summary. Report; UC002ER53(NSN) R53-Part 2. 6 December 1987.

8. Air Force Inspector General. Functional Management Inspection of Transportation Contingency Planning. Report; PN 83-612. 21 March 1983 - 14 February 1984.

9. Air Force Inspector General. Functional Management Inspection of USAF Vehicle Authorization, Acquisition, and Allocations Programs. Report; PN 86-611. 27 January 1986 - 31 December 1986.

10. Van Scotter, Capt James R. Functional Description For A WRM Fleet Management System. AFLMC Report; LT860840. Air Force Logistics Management Center, Gunter AFS AL, July 1987.

11. Department of the Air Force. Logistics War Reserve Materiel WRM Policy. AFR 400-24. Washington: HQ USAF, 28 November 1986.

12. United States Armed Forces Europe. War Reserve Materiel Management. USAFER 400-24. Ramstein Air Base: HQ USAFE, 30 April 1986.

13. Pacific Air Forces. War Reserve Materiel Management. PACAFR 400-24. Hickam Air BASE: HQ PACAF, 9 May 1986.

14. Department of the Air Force. Acquisition, Management, and Use of Motor Vehicles. AFR 77-310; vol I. Washington: HQ USAF, January 1987.

15. Department of Defense. Fiscal 86 - Annual Report to Congress. Washington: Government Printing Office, 4 February 1985.

16. Rowsey, Captain James M. Jr. U.S. Navy. "New Programs Change Operational Role of MSC," Defense Transportation Journal: 20-24 (August 1985).

17. Air Force Logistics Command. World Wide Vehicle Requirements Summary. Report; UC00ZER85(NSN), September 1977.

18. Castro, Luis and Jay Hansen. DBASE Advanced Programmers Guide. Ashton-Tate, 1985.

19. Elbra, Richard A. Database for the Small Computer User. Manchester England: NCC Publications, 1982.

20. Smith, Claire C. Lieutenant Commander U.S. Navy. Supply Hotlist Report Generation For Fleet Balistic Missile Submarine Management Meetings. MS thesis, AFIT/GLM/LSM/87S-70. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1987 (AD-A186677).

21. Capron, Henry L. Computers and Data Processing (Second Edition). San Francisco, CA: Benjamin/Cummings Publishing Company Inc. 1984, 1982.

22. Deutsch, Michael S. Software Verification and Validation. New Jersey: Prentice-Hall Inc. 1982.

23. Simpson, Henry. Design of User-Friendly Programs For Small Computers. San Francisco, CA: McGraw Hill Book Company 1985.

24. Jones, Edward. dBase III Plus Programmer's Reference Guide. Indianapolis, IN: Howard W. Sams & Company 1987.

25. Carrabas, Joseph-David. dBase III Plus Programmer's Library. Indianapolis, IN: Howard W. Sams & Company 1987.

26. MS-DOS Version 3.2 User's Guide and User's Reference. ST. Joseph, MI: Zenith Data Systems Corporation 1986.

27. Seymour, Jim. "Project Database 3 - Covering All the Bases," PC Magazine, 7: 153-254 (12 April,1988).

28. Seymour, Jim. "Project Database 3- Relational Databases: Taking the Middle Ground," PC Magazine, 8: 153-198 (26 April, 1988).

29. Seymour, Jim. "Project Database 3- Programmable Databases: dBase and Its Challengers," PC Magazine, 9: 93-272 (17 May, 1988).

30 Blanchard, Benjamin S. Logistics Engineering and Management (Third Edition). New Jersey: Prentice-Hall Inc. 1986.

31 McGowan, Clement L. and John R. Kelly Top-Down Structured Programming Techniques. New York: Van Nostrand Reinhold Company 1975.

32. Martin, James. Managing The Data-Base Environment. New Jersey: Prentice-Hall Inc. 1983.

Vita

Lieutenant Robert S. Thomas [redacted] He entered the United States Air Force on 5 January 1971, and upon completion of basic military training attended Phase I Medical Laboratory Training at Sheppard AFB, Texas, and Phase II Training at Scott AFB, Illinois. Upon graduation on 18 April 1972, he was assigned at Wright Patterson AFB serving in all sections of the laboratory. He arrived at Hill AFB, Utah in November 1975 and served as NCOIC Microbiology, Clinical Chemistry, and Phase II Training Course Supervisor. He was next assigned to Lages Air Base, Azores from July 1979 to July 1981 and completed an Associate of Applied Science in Medical Laboratory Technology from the Community College of the Air Force. He departed the Air Force at Offutt AFB on 17 August 1982. He attended Weber State College at Ogden Utah, graduating with a Batchelor of Science in Biology in December 1984. He attended Officer Training School in January 1985 receiving his commission on 21 May 1985. Lieutenant Thomas was subsequently assigned to Malmstrom Air Force Base, Montana, serving as Vehicle Maintenance Officer and Vehicle Operations Officer. He was selected as the SAC Transportation Officer of the Year for 1987. He entered the School of Systems and Logistics at the Air Force Institute of Technology in June 1988.

[redacted]

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GLM/LSM/88S-70	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Systems and Logistics 6b. OFFICE SYMBOL (If applicable) AFIT/LSM	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology (AU) Wright-Patterson AFB OH 45433-6583	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
A COMPUTER BASED DATA MANAGEMENT SYSTEM FOR AIR FORCE WAR RESERVE MATERIEL (WRM) VEHICLE MANAGEMENT

12. PERSONAL AUTHOR(S)
Robert S. Thomas, B.S., First Lieutenant, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
MS Thesis	FROM _____ TO _____	1988 September	181

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Data Bases; Computer Programs; WRM Vehicles; Management Information Systems; Transportation
13	06		
12	05		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Thesis Chairman: John M. Halliday, Lt Col, USAF
Head, Dept of Log Mgt

Approved for public release IAW AFR 190-1.

WILLIAM A. MAUER 17 Oct 88
Associate Dean
School of Systems and Logistics
Air Force Institute of Technology (AU)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
John M. Halliday, Lt Col, USAF	(513) 255-5023	AFIT/LSM

DD Form 1473, JUN 86 *Previous editions are obsolete.*

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

ABSTRACT

The purpose of this ~~research~~ thesis was to determine the feasibility of developing a microcomputer based system for use by transportation personnel to manage the War Reserve Materiel (WRM) fleet. This research determined user requirements, developed a prototype system, and validated the prototype system through pre-field testing.

Coordination with Air Force Logistics Management Center (AFLMC) transportation personnel, and HQ PACAF/LGT personnel was established to develop a field testing program for successful implementation of the WRM Vehicle Management System. The prototype system permits manual and automated input from the Vehicle Integrated Management System (VIMS) and the AFLMC Vehicle Automated Management System (VAMS).

The WRM Vehicle Management System provides capability for vehicle dispersal/distribution management, scheduled actions management, release case management, and a variety of reports for the whole fleet or a subset of the fleet. The end product is a computer program on a single 5¼ inch "floppy disk" that will operate on IBM™ or Zenith™ microcomputers. The program was compiled to provide stand alone capability to limit the cost of implementation.

Keywords: management information systems; computer program documentation; (KS)